UNL

平成１５年度
研究開発成果報告書

ユビキタスコンピューティング環境を実現する基盤ネットワークプロトコルの研究開発

株式会社横須賀テレコムリサーチパーク
ユビキタスネットワーキング研究所
研究代表者：坂村　健

目次

添付資料１：研究者発表、講演、文献等一覧

第一章
研究開発課題の背景

20 世紀後半より，情報通信技術・IT（Information Technology）の急速な進展と広範な普及によって，我々の社会は大きく変革し，いわゆる情報社会へと突入した.我が国も情報通信技術に関しては世界を牽引した数少ない国の一つして自負するに十分であり,多くの研究開発がなされてきた．現在も次世代携帯電話を始めとして，世界に貢献する成果を輩出している．

1.1 ユビキタスコンピューティング

1990 年代からの情報通信網基盤の爆発的発展は，ネットワークの大容量化と接続機器（コンピュータ）の高性能化によって，より高度なユーザサービスを実現してきた．それと同時に，近年の我が国では，これとは異なる情報通信網の急速な発展も起きている．それは，従来は情報処理能力や通信能力を持たなかった,身の回りに存在する無数の小さな「モノ」に対して，計算力と通信力を与える方向への爆発的な拡大である.こうした身の回りのあらゆるものをインテリジェント化することで,高いユーザサービスを実現する情報通信のパラダイムは,ユビキタスコンピューティング （Ubiquitous Computing）や「どこでもコンピュータ環境」と呼ばれている．このパラダイムは，1980 年代後半に日米で同時に提唱され，その後ポスト PC 時代の情報通信技術のパラダイムとして，専門家の間で広く受け入れられている．

このユビキタスコンピューティングこそが今後の日本型の IT 技術開発のパラダイムとして有望なものであり,本研究開発課題はこのパラダイムの実現に対して正面から取り組むものである．

1.2 我が国の産業構造との関係

情報通信分野は極めて広範で深いため，単一の国や会社，組織ですべてをカバーすることは，もはや不可能である．世界全体でみた情報通信分野は，様々な国の様々な組織がそれぞれに得意な部分を分担し，世界規模で協調と競争をしながら発展していくのが健全な姿である．

現在，すでにインターネットや PC の分野の技術的主導権は米国が握っている．実際，パーソナルコンピューティング分野は，心臓部である CPU や OS といった根幹技術を"Wintel"という造語が示すように，米国の特定ベンダーの独占状態にあり,我が国の研究開発は壊滅状態である.しかし，情報通信分野で十分に大規模な収益が見込める分野は，インタ

ーネットや PC の分野だけではない．特に，ポスト PC 時代の主力産業と考えられる情報家電，ネットワーク機能をもった電子機器に目を向ければ，ITRON（Industrial TRON: The Real-time Operating system Nucleus）を始めとして，我が国の独自技術が世界的に強い競争力を維持し続けている．我が国の産業構造からみて得意な部分とは，小さく緻密な機器を生産するところにある．こういった視点からも，効率的な研究開発投資を考えると，むしろ，我が国が最も得意な分野を更に発展させることを目指すべきである．こうした技術は今だ世界の先端を走っており，この点を活かした新しい産業を創造し，世界をリードする分野を積極的に開拓すれば，当該分野の世界的イニシアチブを獲得することも可能である．

1.3 緻密でクリーンな IT 技術開発への要求

21 世紀を迎え，光ファイバー網を使ったインターネット等が目指している，より速く・より大容量・より広帯域を追求する情報通信技術の重要性は高いものの，現在それに加えて更に，次のようなより緻密でクリーンな情報通信技術への要求も高まっている．第一に，豊富な容量・帯域・速度をもった IT 基盤上のデジタル情報の流れを，人間や社会の意志に基づいて確実に制御できること．権利がある人にだけに情報のアクセスを許可し，権利の無い人の不正な盗聴を防ぐこと，また，不正な情報複製ができないようにするといったことが，確実に行えることが重要である．従って，これには，広い意味での，暗号技術や認証技術などが含まれる．第二に，省電力をはじめとして，資源を浪費せず，環境への悪影響を最小限にとどめる，クリーンな情報通信技術．こうした考え方は，カームコンピューティング（Calm Computing）とも言われる．

1.4 セキュアコンピューティング

2001 年は，我が国でもサイバーテロが行なわれた．また，Nimda，CodeRed，Circum といったインターネットを介して大規模に感染するコンピュータウィルスやワームも発生した．既に米国では我が国以上にサイバーテロが行なわれ，情報社会を脅かしている．今後もこうした危険性は確実に拡大するだろう．現在の情報通信インフラで解決することが最も要求されている課題は，こうした攻撃に強い，セキュアな情報通信基盤である．しかもそれが，一般素人でも簡単に扱うことができなければならない．インターネットの当初の設計方針には，インターネットを

通じて通信する者同士が信頼できないようなこと，また，これほどまでに素人ユーザの割合が多くなることは，想定されていなかった．現在これに対して抜本的対策を施さない限り，かつての公害問題のように，将来の情報社会に禍根を残すことになりかねない．

第二章
研究開発分野の現状

身の回りのあらゆるものにコンピュータをうめこみ,それらが互いに協調動作することによって高い機能とサービスを提供する情報通信環境は，Ubiquitous Computing という言葉で代表され，1980 年代～1990 年代にかけて，日米で提唱された.

米国では Xerox 社 Palo Alto Research Center (PARC) の Mark Weiser 博士が,自らの研究グループによるユビキタスコンピューティングの研究を発表した，1991 年 9 月の Scientific American 誌の論文"The Computer for the 21st Century"が，最初にユビキタスコンピューティングのコンセプトを提案したもので,これは特にユーザインタフェースの研究者に対して大きな影響を与えた.というのも，当時のユーザインタフェースの研究は,コンピュータグラフィックスを活用した GUI の研究や，マルチメディアを使ったインタフェース研究が主流であり，どちらもデスクトップのコンピュータと相対して対話するものであった.ところが Weiser 博士は，非デスクトップのユーザインタフェースの重要性を説き，むしろそちらの方が，通常の人間生活の中で接する頻度の大きいインタフェースであることを述べた.まだ当時は，新しい技術が開発されたというよりは,当時の技術を利用してユビキタスコンピューティングが目指す姿をとりあえず構築し,そのユーザインタフェース上の有効性を検証するものであった.日本では，本研究課題の研究代表者である坂村健が，1989 年に超機能分散システム（HFDS: Highly Functionally Distributed System）としてこのユビキタスコンピューティングのコンセプトを提唱した.PARC の研究よりも，より分散システム技術に比重が置かれている点が特徴である.

2.1 Xerox PARC の UbiComp［1991～］

Xerox Palo Alto Research Center (PARC) では，Mark Weiser 博士を中心としたグループによって，Ubiquitous Computing (UbiComp)の研究が行われていた.UbiComp は，人間の生活のいたるところで，人間の目には触れないコンピュータを利用する環境の構築を目指している.UbiComp は，以下の様なシナリオの実現を目指していた.

- オフィスビルの従業員は，常にアクティブバッヂを身につける.
- 正当なアクティブバッヂを身に着けている時しか開かない自動ドア.
- 名前で挨拶してくれる部屋.
- どこに居ても居場所にかかってくる電話.

- 従業員がどこにいるか把握している受付.

このシナリオを実現することに焦点が置かれており,それを実現する上で必要な通信プロトコルや，リアルタイム性，セキュリティー，運用や利用の容易性，更にはコンパクトな実現方法などには，ほとんど取り組まれていない.あくまでも技術的な課題を抽出するためのフィージビリティー研究に近い.

2.2 RANK Xerox EuroPARC

RANK Xerox EuroPARC でも， Pierre Wellner 博士を中心とした研究グループが,早い時期からユビキタスコンピューティング研究に取り組んできた. 代表的な研究は DigitalDesk といわれるもので，実物の机をコンピュータのモニタのように使ってコンピュータと対話する技術である.従来の GUI がデスクトップを模擬してコンピュータの画面を構築していたケースと逆に,本物のデスクトップをコンピュータとの対話の場に使うものである． これはユビキタスコンピューティング

環境の一部として使うことはできるものの,ユビキタスコンピューティング研究の中でも,ユーザインタフェース部分に限定されたミクロレベルの研究である.

2.3 MIT Media Lab.

MIT Media Lab.では， TTT （Thing That Think）という研究プロジェクトがあり,ユビキタスコンピューティング環境への研究に取り組んでいる. 代表的な研究として，コンピュータを埋め込んだインテリジェントなおもちゃであるとか,洋服を「着る」感覚で常に携帯する「Wearable Computer」の研究がある． TTT の上位コンセプトとして「Tangible Bit（触れるビット)」というものがあり， コンピュータやネットワーク上の仮想的なデジタル情報に対して,実世界のモノを通してアクセスするというコンセプトを実現するために,身の回りのものにコンピュータを埋め込んでいるのであって,我々がユビキタスコンピューティングで目指している目標と方向性が異なっている.

2.4 MIT AI Lab.

MIT AI Lab.では， SmartRoom という研究が行なわれている． 部屋の中にコンピュータを埋め込み,部屋が知的に振舞うようにすることが目的である． ここは， 人工知能の研究所であることから，研究の力点も，

いかに部屋を「知的」に振舞わせるかという部分に重点がおかれており，あくまでも人工知能研究の一貫として取り組まれている．

2.5 IBM Pervasive Computing Project

IBM の Pervasive Computing Project では，ポスト PC 時代に向けた IBM 社の戦略の一環として進められているプロジェクトである．従来，大型計算機や PC を商品としてきた IBM 社が，情報家電といた非 PC 製品を商品化するために必要な技術開発をしており，本研究開発課題のような，基盤技術を扱うものではない．

2.6 US NIST Pervasive Computing Project

米国の NIST にも Pervasive Computing を扱うプロジェクトがある．ここの特徴は，現在ネットワーク型言語として広く使われている Java 言語の処理系をコンパクトかつ軽量化することや，Java にかわるよりコンパクトで実行効率の良いネットワーク言語の開発，またその上のミドルウェアなど，やはり製品改良に近いレベルの研究が行なわれており，本研究開発課題のような基盤性を有する研究ではない．

2.7 SONY CSL

国内では，SONY Computer Science Laboratory (CSL)の Interaction Laboratory の暦本氏らによって，"Augmented Interaction"の研究がなされている．ここの研究も研究室レベルの小規模なものであること，また，あくまでも情報環境との Interaction を研究しており，我々のように，基盤プロトコルに関する研究開発を行っているわけではない．

2.8 UCB／Endeavour ［1999～2002］

カリフォルニア大学バークレー校(UCB)の電気電子および計算機科学科では，Randy H. Katz 教授を中心とした多数のグループによって，コンピュータが内蔵された多種多様な情報機器（大型コンピュータから，家電製品，さらには超小型センサーまで）を相互接続することにより，あらゆる種類の情報を収集し，さらにその情報を即時に引き出したり，情報に従って最適な対応を自動的に行ったりできるような次世代コンピュータネットワークを構築するのに必要となる要素技術の研究･開発が進められている．これにより，例えば家に取り付けられた情報収集機器により，その家の住人の生活パターンを自動的に認識し，その収集さ

れた情報に従って各機器が動作する Smart House 等が実現できる．このために，

- 超小型センサー
- クラスターベースの大規模計算処理及びメッセージ処理システム
- スケーラブル，メインテナンスフリーのストレージサーバーシステム
- ネットワーク上で，スケーラブルかつ安全にサービスを実行できる環境
- 異種の通信デバイス間で広域通信やモビリティを提供するプラットフォーム
- 超小型デバイスのための OS
- 広域データ蓄積システム
- 適応型データフローシステムの開発
- 広域ネットワーク上でデバイス間の通信を動的に制御（帯域管理，複数コネクション間の同期確保，動的な接続の切断・再接続，等）するための機構
- セキュリティツール

の研究が行われている．基本的には，各サブプロジェクトが独自に研究を進めているため，プロジェクト終了後に新しいコンピュータネットワークシステムが完成するというものではない．また，研究の内容については，新規のコンピュータネットワークを作るというよりは，現在のインターネットに導入できるような技術として検討している面がある．これらの点において，新規にユビキタスネットワークを研究開発することを目的とする本研究課題のアプローチとは異なっている．

2.9 MIT／Oxygen ［2000～2005］

遍在する自己組織化するコンピュータ網を実現するために MIT の LCS と AI Lab が共同で DARPA の資金により進めている統括プロジェクトで，LCS の所長 Victor Zue と副所長 Anant Agarwal および AI 研の所長 Rodney Brooks が中心となってすすめている．この目的を実現するために必要となる個々の技術要素，例えば，知識の格納及びアクセス，網構成の自動化，網構成要素間の強調，音声認識，画像認識，変更可能なソフトウェア，N21 と呼ばれる分散型ネットワーク，センサー等が接続される E21 と呼ばれるデバイス，H21 と呼ばれるユーザ用のハンドヘルドデバイス，等 30 ものサブプロジェクトにおいて研究が進められている．既にいくつかの E21，H21 の試作が発表されている．但し，ネットワークに関し

ては Endeavour と同様に従来のインターネットをベースとしており,本研究課題のアプローチとは異なっている.

2.10 CMU／Aura ［2000～］

CMU は，計算機学科の学部長 Raj Reddy を中心に，ユーザの注目が最も重視すべき資源であるというコンセプトの元に,ユーザに気を散らせない,高信頼なモバイルネットワークとその上のサービスを提供するための研究を Aura と呼ぶ統括プロジェクトの元で進めている．タスク指向型コンピューティング，省電力コンピューティング，ウエアラブルコンピューティング，音声認識，分散協調，等を課題に 9 つのサブプロジェクトで研究が進められている.基本的に Endeavour と同様に既存のネットワークの上を考慮した QOS 制御やルーティングなどを研究しており，本研究課題のアプローチとは異なる．また，リアルタイム OS の研究もマルチメディア通信を提供するためのもので,本研究の小型かつセキュアなリアルタイム OS の研究とは方向が異なる.

2.11 ワシントン大学／Portlano［2000～］

ワシントン大学は，XEROX PARC と共同で，ユーザに，計算機の存在を意識させない分散サービス提供環境を実現するために必要となる,多様なユーザインタフェースの統合技術,自律的にデータを分散格納しそれに対するアクセスを提供するロバストなデータ指向のネットワーク技術,およびその網の上で多様な分散サービスを提供するためのソフトウェア技術の研究を進めている.具体的には,大型ユビキタス表示装置,新しい位置センサー，モジュール式のセンサー，安価な ID システム,センサーの統合技術，プロトタイププラットフォームの試作及び，試作プロトタイプの大規模実証実験等 15 のサブプロジェクトにより進められている．主に，ネットワーク上でのデバイスの開発と分散処理技術に重点があり，本研究課題で提案するような，セキュアかつリアルタイムな通信とそのために必要となる OS やプロトコルの開発は研究の対象外となっている.

第三章
研究開発の全体計画

3.1 研究開発課題の概要

本研究開発課題は，我々の身の回りの，あらゆるものにマイクロコンピュータと通信機能を組み込み，それらが互いに情報を交換しながら協調動作を行い，人間生活をより高度にサポートする，ユビキタスコンピューティング環境を構築するための，次世代通信の基盤プロトコルおよびそのシステムを確立することである．

3.1.1 ユビキタスコンピューティング環境が目指す最終目標

ユビキタスコンピューティング環境とは，身の回りのあらゆるものにマイクロコンピュータと通信機能を組み込み，それらが互いに情報を交換しながら協調動作を行い，人間生活をより高度にサポートする環境のことである．今まで，こうした環境を使った IT の多様な「夢」が語られており，その「夢」を実現することが本研究プロジェクトの最終目標である．

例えば，家庭では，家に設置された温度センサーが常に外気温と室内温度を監視しており，居住者が部屋の温度を下げようとした時，もしも外気温が室温より低ければ窓を開ける．しかし，部屋でピアノを弾き，外部に騒音が漏れたら，窓を閉めて自動的に空調が入る．また，自家用車で帰宅するときに，自動車のナビゲーションシステムから，到着時刻に合わせて自宅の風呂の湯を沸かすこともできる．

こうした環境を実現するためには，膨大なコンピュータを身の回りのあらゆるものに埋め込み，人間自身も常にコンピュータを携帯し，それらが互いにネットワークで接続され，情報交換しながら協調処理を行うメカニズムが必要とされる．我々は，この埋め込まれたり携帯されるコンピュータを**「インテリジェントオブジェクト」**とよび，これらを接続する通信メカニズムのことを**「ユビキタスネットワーキング」**と呼ぶ．

本研究では生活空間を構成する大量のインテリジェントオブジェクトからなるネットワークを想定している．このネットワークと他の既存のネットワークとの違いは，まずネットワークにつながるノードの数が桁違いに多いことである．一人当たり数十から数百のプロセッサがある高密度のユビキタスコンピューティング環境のなかから，通信すべき適切なコンピュータを指定するためにはどうすればよいか．更にそれが何百，何千もの人が活動するビルや都市，最終的には世界までつながった時に，このユビキタスネットワーキングがどのように展開されるべきか，といったことが重要な課題となっている．従って，本研究における中心

課題は，この「ユビキタスネットワーキング」の根幹となる基本方式を明らかにし，更にそれを動作させるシステムを構築することである．

これらの多くのインテリジェントオブジェクトを協調させるためには，調停動作の実現がポイントである．例えば，一億個のコンピュータがネットワークにつながった場合，全部のデータを手に入れそれに基づいて中央で方針を決定するという，中央集権的な手法で全部の動作を最適化することはもはや不可能ではないかと考えている．そのためには何らかの分散的な最適化方式を考案する必要がある．

生活の場におけるインテリジェントオブジェクトは個々のエンドユーザの都合でネットワークに突然追加されたり，はずされたり，次の日には別の箇所につながったり，ということが起こる．そのような**「アドホック（ad hoc）」**性をもったネットワークを実現しなければならない．その際に一般の人でも扱え，面倒なオペレーションが不要な**「エフォートレス（effortless）」**な性質を持つ必要がある．ユビキタスコンピューティング環境において，無数のコンピュータをちりばめた時に重要なことは，その上で，これらのコンピュータ群が 24 時間 365 日正常動作するように運用できることである．そのためには，ユビキタスコンピューティング環境を構成するシステムと社会との親和性，運用技術に対する研究も重要となる．

3.1.2 技術課題の概要

本研究では，このユビキタスコンピューティング環境の基盤技術となる通信プロトコル（**ユビキタスネットワーキングプロトコル**）や，それを用いた通信網基盤の構築技術の研究開発を行う．そのために以下の研究開発項目を実施する．

1. リアルタイム通信プロトコル
2. セキュアネットワーキング，セキュアコンピューティング
3. コンパクト性
4. エフォートレスオペレーション，エフォートレスマネジメント
5. ユーザとの親和性
6. 省リソース
7. 既存通信網との親和性
8. 高度な協調・調停動作による人間生活の支援機能の実現

（1）リアルタイム通信プロトコル

ユビキタスコンピューティング環境を実現するための基本プロトコ

ルには，人間の振る舞いや生活・社会を構成するあらゆる事象に追従して応答できるための，（ソフト）リアルタイム性が必要である．特に，身の回りのインテリジェントな機器（アクチュエーター）を制御する部分には，より強いリアルタイム性が要求される．

（２）セキュアネットワーキング，セキュアコンピューティング

先に述べたユビキタスコンピューティング環境による夢，例えば，未来の ITS（Intelligent Transportation System）のイメージとして，自動車のナビゲーションシステムから，到着時刻に合わせて自宅の風呂の湯を沸かすといったシナリオが描かれてきた．実際に，このシナリオを実現するためには，悪意ある他者が自宅の風呂を操作することを防げなければならない．更に近年は，サイバーアタックやクラッキング，サイバーテロを想定した対策も求められている．ユビキタスコンピューティング環境を，ネットワーク経由のアタックから守るためには，セキュアな通信プロトコル，セキュアな通信システムの開発が不可欠である．ネットワーク基盤の遍在化（ユビキタス化）が急速に進んでいる現在，セキュリティーを向上させる技術開発は急務である．

（３）コンパクト性

ユビキタスコンピューティング環境では，非常に膨大な数の小さな機器にコンピュータや通信機能が埋め込まれる．従って，各ノードが小さくコンパクトであること重要である．計算機能や通信機能の偏在性（ユビキタス性）を高めるためには，各ノードがコンパクト化できることが不可欠である．

（４）エフォートレス（Effortless）

ユビキタスコンピューティングのためのネットワークプロトコルを実現する上で重要なことは，コンピュータに詳しくない一般利用者でさえも，自身が所有する機器の安全性，蓄積情報や通信の秘匿性等を手軽に確保できることである．現在でも多くのセキュアプロトコルがあるが，そのほとんどが，認証や暗号のための鍵の管理や認証局（CA 局）からの証明書の取得といった，専門知識を要する運用作業を伴うため，普通の人が手軽に使えるものになっていない．そこで，本研究課題では，耐タンパ性を有するハードウェアを利用することによって，より手軽で簡便なセキュア通信プロトコルを開発する．このプロトコルによって，簡単に使えるパッケージ化された強固で安定した汎用セキュリティー基

盤が実現され，コンピュータに詳しくない普通の人でも，セキュアな通信基盤の恩恵に浴することができる．

（５）ユーザとの親和性

ユビキタスコンピューティングは，人間の身の回りに計算機能や通信機能を埋め込むことで，人間生活をサポートする．そこで，重要な観点の一つは，どのように人間をサポートしていくかということである．ユビキタスコンピューティング環境においてこうした人間との境界部分，つまりヒューマンインタフェースをつかさどる分野は，**強化現実環境**（Augmented Reality）とか，**複合現実環境**（Mixed Reality）といった分野となる．本研究でも人間にとって，より自然でかつ利便性の高いサポートの手法の研究開発を進める．

（６）省リソース

ユビキタスコンピューティング環境では，ノード数が従来の分散環境やインターネット環境にもまして膨大な数になることから，一つのノードが使う電力量など，消費する各種リソースが小さくなるような Calm Computing 技術を確立する．従来の情報通信技術は，性能や利便性を最大化するための研究開発に重きがおかれており，省資源を最大化するといった基準に則った研究開発は比重が低かった．本研究課題で実現するプロトコルやシステムでは，こうした省電力・省資源に取り組む．

（７）既存通信網との親和性

現在，IP による世界的ネットワークが構築されている．その他にも既存の通信網には，携帯電話網，通常の加入電話網をはじめとして，多様なものが存在する．これらは，性能，サービス，保守の容易性といった点で利害得失があり，これらが単一のプロトコルに統合されるとは考えづらい．ユビキタスコンピューティング環境を構成する通信プロトコルに関しても同様に，様々な利害得失をもった多様なプロトコルが混在する環境を前提とし，互いに補完的な関係になることが理想的である．

そこで，本研究開発課題では，こうした既存の多様なプロトコルとの相互接続によって，柔軟な情報交換や制御を行うメカニズムを確立する．具体的には，IP に基づくインターネット，携帯電話網上に構築されている情報通信網基盤と，ユビキタスネットワーキングプロトコルとを相互接続する，ゲートウェイ技術を確立する．それは単に，ネットワーク層やトランスポート層によるゲートウェイだけではなく，セッション層

やアプリケーション層にいたる,ほぼ全ての層において接続する必要があり，そのための統合的なゲートウェイ技術を構築する.

（８）高度な協調・調停動作による人間生活の支援機能の実現

ユビキタスコンピューティング環境では,単に多くのノードがあるだけでなく，それらが「協調」「調停」動作をすることで，人間生活を高度に支援する．例えば，部屋の温度が上ったら，窓を自動的に開け，もしも部屋の中でピアノをひきはじめ騒音が発生したら,窓を自動的に閉めて空調を入れるといった動作である．こうした協調・調停作業が，あちこちに埋め込まれた膨大な数のコンピュータの間で実施できなければならない．そのための通信システム，交換情報形式，超機能分散型ソフトウェアのためのプログラミングシステムなどの研究開発を行う.

3.1.3 研究実施計画の基本方針（概要）

ここでは，2.1，2.2 で示した本研究の目標と課題を達成する実施計画の方針の概要を示す.

（１）システム単位のサブプロジェクト構成

本研究開発課題における技術的ボトルネックは,いかに大量の小さいノードを，特定の目的に沿って協調動作させるかという，システム統合技術にあると考えている．そこで，要素技術毎に細分化したサブテーマわけをした場合,最後に統合化して相互接続環境を構築するあ困難になるため，次の通り「機器」による切り分けを中心としたサブテーマわけを行う.

1. セキュアハードウェア
2. 通信システム
3. ユーザ側のエンドノードシステム（モバイル端末／情報家電／PDA等）
4. サーバ側のエンドノードシステム（認証サーバ／アプリケーションサーバ等）
5. システム統合技術
6. 超機能分散システム指向の開発環境

（２）システム工学的検証の重視

ユビキタスコンピューティング環境は,単に情報通信のメカニズムだ

けを研究開発するだけでは成功しない.ユビキタスコンピューティング環境は，人間・社会生活に無数に埋め込まれ，社会活動や生活を支援するものであるため，技術的に優れていても，例えば騒音や発熱の程度によっては人間生活にはなじまない．公共の場に設置するものは，多少の悪戯や荒い扱いにも耐えなければならない.膨大な数のノードが現実社会の中で容易にかつ安全に運用できるのか,無数に埋め込まれたインテリジェントオブジェクトのメンテナンスは可能なのか,ユビキタスコンピューティング環境のセキュリティーは厳密に運用できるのか,といった諸問題がある.

そこで，本研究開発課題では，こうした社会や生活と，ユビキタスコンピューティング環境の間の親和性を重視し,本研究開発成果を実用に耐えるシステムとして完成させるために,システム工学的見地からその検証を行う．検証項目としては，①システムの信頼性の検証，②運用評価，③ユーザビリティ，④スケールファクターのシミュレーション，⑤環境アセスメントを計画している.

3.1.4 研究実施計画の詳細（システム部分）

システム面の研究のサブテーマは，「2.3 （1）システム単位のサブプロジェクト構成」の部分で述べたとおり，機器種類毎に切り分けを中心とする．サブテーマとしては，以下を計画している.

【サブテーマ１】
セキュアコンピューティングの基盤となるセキュアハードウェア

【サブテーマ２】
基盤通信システムの研究開発

【サブテーマ３】
ユーザノードシステムの研究開発

【サブテーマ４】
サーバノードシステムの研究開発

【サブテーマ５】
ユビキタスコンピューティング環境を構成するシステム統合技術の研究開発

【サブテーマ６】
超機能分散システム指向の開発環境の研究開発

【サブテーマ１】
セキュアコンピューティングの基盤となるセキュアハードウェア

通信のセキュリティーは一般的にはソフトウェアだけで確保することはできない．何らかのハードウェアによる情報保護が不可欠である．従来型の情報システムでは，機材が設置されている建物や部屋に対する入退館管理等による物理的な保護が前提であった．ユビキタスコンピューティング環境では，公共の場に露出して設置されたものも対象であり，ユーザが常に携帯し頻繁に紛失や盗難が起きるものまで含まれる．しかも，前述したようにセキュリティーを確保するために，ユーザが暗号・認証の仕組みを理解することが必須であってはならない．

そこで，本研究では，LSI 自体に不正アクセスが加えられないように加工を施した，いわゆる**「耐タンパー性（Tamper Resistance)」**を有するハードウェアを，ユビキタスコンピューティング向けチップとして新規に開発し，それをユビキタスコンピューティング環境のセキュアシステムの基盤パーツとする．

【サブテーマ２】
基盤通信システムの研究開発

(2-1) 基盤プロトコル概要

基盤通信システムのサブテーマでは，ユビキタスコンピューティング環境を構成する通信システム全般を扱い，本研究の核であるユビキタスネットワークプロトコルの研究，そのプロトコルスタックの開発，ルーターをはじめとした各種ネットワーキング装置を含む．ユビキタスコンピューティングシステムにおいては，その目的に最適化したネットワークアーキテクチャを導入する．現時点では，研究対象となるユビキタスネットワークのアーキテクチャと機能は以下の特徴をもつものを想定しているが，これは研究の進捗・進展や，他の技術動向に応じて変化する部分もある．

(2-2) データリンク層

データリンク層は既存の方式を用いる．例えば，2.5GHz および 5GHz 帯の無線 LAN，Bluetooth，ISO 14443，PHS，第 3 世代の移動体通信ネットワークなどである．これらの方式としては，端末と端末が直接通信するアドホックモードの通信形態と，基地局およびバックボーンネットワークを介した通信形態の双方を検討する．

(2-3) ネットワーク層

ネットワーク層はユビキタスネットワークに適した新しい方式を考案して実現する．通信形態は，ユニキャストとマルチキャストをサポートし，それぞれ帯域保証や優先制御を行うリアルタイム通信と，ベストエフォート通信を扱う．帯域保証等を行うリアルタイム通信の場合は，そのためのシグナリングを提供する．

ここでは，Layer 2 ARP（Address Resolution Protocol）が必要である．ユビキタスネットワーキングでは，IP で使用される ARP とは異なり，実世界上の位置や社会的なセマンティックスなどに基づいたノード指定に対応する（デ**バイス・ノードルックアップ機能**）．

ネットワーク構成やノード間の帯域などのルーティング情報を交換するルーティングプロトコルを実現する．このプロトコルは，ユニキャストのためのプロトコルと，マルチキャストのためのプロトコルの双方，またダイナミックな変化に追従できる柔軟性が必要となる．

更に，ネットワークにおける輻輳や障害等を通知するための OAM (Operation Administration and Maintenance)情報交換プロトコルを備える．このプロトコルは，ネットワーク経路上の故障に加え，リアルタイム通信のための輻輳の検知やマルチキャストにおける障害情報の転送などを，統合的にサポートする．

(2-4) トランスポート層

トランスポート層のプロトコルとしては，コネクション型とコネクションレス型のプロトコルを用意する必要がある．双方のプロトコルとも，遅延変動や誤り率などのサービス品質に関して，アプリケーションが要求する品質を最小限の機能で実現するサービス品質機能を有する．コネクション型のプロトコルはユニキャストを対象とし，コネクションレス型のプロトコルはユニキャストとマルチキャストの双方を対象とする．また，通信相手の指定については，アドレスを指定する方式のほかに，要求条件を指定する方式についてもサポートする．確認応答を有し，信頼性の高い通信を可能とする．

(2-5) セッション層

セキュリティーや認証のための機能を提供する.相互認証と同時に鍵交換を行い,セッション中は暗号通信が行なわれるセキュアセッション機能,またロールバック機能を備えたトランザクションセッション機能,リアルタイム応答が要求される場合のライトウェイトセッション機能を備える.

(2-6) アプリケーション層

アプリケーション層は，各種応用に対応するプロトコルを用意する.その他に,通信のサポートのために各種応用から共通に使用されるプロトコルを用意する必要がある．1 つは，サービスルックアップ機能で,サービス名からそれを提供するノードの物理アドレスを検索するなど,各種のネットワーク構成情報の検索プロトコルを構築する(**サービスルックアッププロトコル**)．また，ネットワーク機器の状態監視や構成変更などを遠隔で行うネットワーク管理用プロトコルも備える.

【サブテーマ３】
ユーザノードシステムの研究開発

ユビキタスコンピューティング環境を構成するノードの中で,ユーザと直接接することが想定される機器類の研究開発である.これをここではユーザノードと呼ぶが,想定されるユーザノードには,ユーザが携帯する移動ノードと,生活環境に設置される固定ノードがある.双方とも,**【サブテーマ２】基盤通信システムの研究開発**で開発された,ユビキタスネットワーキングプロトコルを搭載する.移動か固定かに応じて,利用可能な計算・通信資源や,物理通信路の環境,物理的な大きさ,それに基づくユーザインタフェース等が異なるために,それぞれ固有の実現技術が必要とされる.本サブテーマでは,こうした条件に適合した様々なユーザノードの構成方法・機構を中心に研究開発を進める.

(3-1) 移動ノード

移動ノードや,常にユーザが携帯してユーザとユビキタスコンピューティング環境との間のインタフェースの役割を担う端末である.具体的には，PDA (Personal Digital Assistant)，携帯電話スマートカードなどが想定される.固定ノードと比べた場合,移動ノードに関する研究課題として以下がある.

- 物理通信路は基本的に無線通信であり，ユーザの移動を考慮すると通信品質は安定しない，
- 紛失・盗難が起こる可能性が高いため，それに備えたセキュリティーメカニズムを備えること，
- 物理サイズが小さいため，それに適した洗練されたユーザインタフェース，
- 計算資源も限られているため，コンパクト性が求められる．
- バッテリー量の制約も大きいため，徹底した省電力機構．

移動ノードでは，こうした条件をクリアした中で，ユビキタスネットワーキングプロトコルの実現技術を確立する．

(3-2) 固定ノード

固定ノードは，ユビキタスコンピューティング環境に設置され，人間の振る舞いや，環境の変化に応じて柔軟かつ緻密に動作するインテリジェントオブジェクトである．具体的には，住宅内にある電子機器類，オフィスにあるコピー機やファックス，シュレッター―といった機器，また公共の場に設置された券売機，自動販売機，チケットゲートといった機器を想定している．移動ノードと比べた場合の固定ノードに関する研究開発項目の特徴は，以下の通りである．

- 物理的認証をうけない不特定多数によって利用されることが前提となり，そのための認証機能，セキュリティー機能が必要とされる．
- 特に公共の場に置かれた機器は，ネットワーク的にも公共的なセグメント上に置かれることになり，その場合にセキュア通信の確保が必要である．
- ユーザノードであると同時に，サーバ的な機能の提供も求められる．
- 回線や電源の状況は良い環境にある．

固定ノードでは，こうした特質を前提とした，ユビキタスネットワーキングプロトコルの実現技術を確立する．

【サブテーマ４】

サーバノードシステムの研究開発

サーバノードは，ユビキタスネットワーキングプロトコルを実現し，特にユーザノードを対象としてネットワークサービスと機能を提供するノードである．具体的には，①セキュアセッションのための認証局，

登録局，②分散トランザクションを支えるトランザクションサーバ，③サービスルックアップやデバイスルックアップのためのネットワーク環境サーバ，④ネットワーク管理のための管理サーバ，⑤各アプリケーションに依存したアプリケーションサーバなどが想定される．

従来の電子商取引分野の研究開発の経験により，サーバの運用者側の不正行為も問題とされており，サーバ側も耐タンパー性を持ったハードウェア構成が必要である．かえって移動型のユーザノードのように小型機器の方が，対タンパー性を実現することが容易であり，サーバノードの場合は，大きなノードにおいての耐タンパーハードウェアの実現技術が課題である．

また，ユビキタスコンピューティング環境においては，サーバにおいても，セキュア性を保ちながら，リアルタイム性を実現できるに十分な高応答性能を実現する必要がある．そこで，本研究開発課題では，サーバをセキュアに性能向上させるための，セキュアクラスタリング，暗号・認証処理機能を持ったデータキャッシュ・プロセスマイグレーションのメカニズムを確立する．特に，動的な情報更新が可能な高応答性を達成できるディレクトリサーバを実現する．

【サブテーマ５】

ユビキタスコンピューティング環境を構成するシステム統合技術の研究開発

ユビキタスコンピューティング環境は，様々なプロトコルを用いる様々な膨大な機器がヘテロジニアスに結合したシステムであり，それを統合し協調動作させるための技術を確立する．その統合技術を要素技術に分割すると，以下の項目が挙げられる．①通信環境を意識する必要のない確実なコネクティビティおよびサービスの実現，②ネットワーク運用状況，サービス実行状況に応じた，最適なリソース配分によるコンパクトネットワーク／サービス実行環境の構築，③通信環境に最適化したサービス実行メカニズム，④ネットワークへのノードの簡単な装着／脱着と移動時のサービス継続の開発．

上記の課題を実現するときに，本研究開発課題で開発するプロトコルと既存の IP や電話網を使った通信プロトコルの相互運用をスムーズに行うため，次の基本技術の開発を目指す．①アドレスを陽に指定しないアドレス解決およびルーティングメカニズム，②プロトコル変換メカニズム，③複数の異なるネットワークをまたがったときの品質制御メカニ

ズム，④端末と連携したサービス実行メカニズム．

【サブテーマ６】

超機能分散システム指向の開発環境の研究開発

ユビキタスコンピューティング環境を構築する上での技術的な課題として，システム開発効率がある．ユビキタスコンピューティング環境は，他の分散環境と比べると，膨大なノード数に特徴がある．現在の計算機科学では，ここまで分散化された多数のノードを協調動作させるソフトウェアを効率よく開発する手法が存在しない．しかも，各ノードはリアルタイムプログラミングとセキュリティーという，単独でも困難なプログラミングを施さなければならない．従って，ユビキタスコンピューティング環境のソフトウェアの開発環境は重要であり，本研究の成否を左右する大きな課題である．

現在計画している開発環境研究は，以下の 3 段階で進める．

1. ユビキタスコンピューティング標準開発環境
2. ユビキタスネットワーキングプロトコルを扱うミドルウェア
3. ユビキタスコンピューティング環境における情報処理モデルの確立

(6-1) ユビキタスコンピューティング標準開発環境

ユビキタスコンピューティング環境は膨大な数のノード数になる．まずハードウェアやオペレーティングシステムといった基盤ソフトウェア部分に対する標準化が必要である．膨大なノード数をまちまちの開発環境で構築しては，ソフトウェアの再利用性の観点から効率よくプロジェクトを運営できない．そこで，まず本プロジェクトの標準ハードウェア，標準基盤ソフトウェアの仕様を決め，その上でのユビキタスコンピューティング環境やユビキタスネットワーキングプロトコルのソフトウェアの再利用性，移植性の高い環境を構築する．

(6-2) ユビキタスネットワーキングプロトコルを扱うミドルウェア

ユビキタスネットワーキングのアプリケーション層のプログラミングを支援するためのミドルウェアを構築する．

(6-3) ユビキタスコンピューティング環境における情報処理モデルの確立

ユビキタスコンピューティング環境を実現する上で最も重要な分散

協調動作を実現するソフトウェアの構築手法を研究する部分である．このテーマにおいても，次の２つのサブテーマを計画している．

1. 現実世界の記述方式
2. 超機能分散環境に適したプログラミングモデル（協調動作の記述）

ア．現実世界の記述方式

ユビキタスコンピューティング環境では，現実世界にコンテクストを取得し，協調動作の結果として，何らかの作用を現実世界にフィードバックする．こうした処理をコンピュータが扱うためには，現実世界をデジタル情報で表現し，それをノード間で交換できるための標準形式を構築する必要がある．しかもユビキタスコンピューティングが扱う事象は，単にオフィス空間といったものだけでなく，人間社会生活のあらゆる場面に及ぶため，まさに，現実世界あのあらゆる事象の標準デジタル表現形式を研究開発する．

イ．超機能分散システムのプログラミング技法

従来は，ネットワーク接続された複数のノード間における分散処理は，オブジェクトベースでモデル化したソフトウェア開発が主流になっている．しかしユビキタスコンピューティング環境のようにノード数が膨大である場合，扱うオブジェクト数も膨大になるため，その間の協調動作をノードオブジェクトレベルの peer-to-peer の協調関係をベースとした動作でプログラミングしては，抽象度が低すぎるということが問題となっている．従って，本研究では，ユビキタスコンピューティング環境全体に対して「計算場（Computing Field)」と呼ばれる仮想的なプログラミング抽象を提供し，各ノードとこの計算場の間の協調動作によってプログラミングする．

3.1.5　研究実施計画の詳細（システム工学的検証）

ユビキタスコンピューティング環境と人間社会･生活の間の親和性を重視し，本研究開発成果を実用に耐えるシステムとして完成させるために，システム工学的見地から，以下の検証を行う．

【サブテーマ７】
ユビキタスネットワーキングシステムのシステム工学的検証

(7-1) 信頼性の検証

本研究開発課題で構築されたシステム（以下，本システム）を，実際のユビキタスコンピューティング環境と同様の設置状況において運用し，そのシステム信頼性を検証する．ユビキタスコンピューティング環境は，生活のあらゆる面を支援するタイプのシステムであるため，誤作動などは致命的であり，この点に関する検証を行う．

(7-2) 運用評価

実際に本システムに想定される技術レベルの人員が，実験的に作られた本システムの環境を，一定期間オペレーションすることによって，①統合的視点によるセキュリティー強度の検証，②運用やメンテナンスの容易性を評価する．

(7-3) ユーザビリティ評価

本システムが利用者に提供するサービスのユーザインフェース手法について検証，評価する．特に，情報通信に関する技術に明るくない一般ユーザに対するユーザビリティ，また，身体障害者や子供，老人を含めたあらゆる人に対して使えるシステムになっているかという，ユニバーサルデザインの視点による評価を重要視する．

(7-4) スケールファクターのシミュレーション

本研究開発プロジェクトはあくまでも研究段階のものであるため，本研究成果が実際に世の中に大規模に普及した場合，どのような問題が起こっていくかを，本研究における実験のサンプルデータを使ったシミュレーションによって検証する．

(7-5) 環境アセスメント

本システムを生活環境に埋め込んだ場合の，放熱，騒音，電磁波などの影響を計測し，人体や他の機器，環境に対する影響を調査する．その結果をシステムの省資源部分にフィードバックしていく．

3.2 研究開発目標

3-2-1 最終目標

■全体を包括する最終目標（概要）

（１） 人間の振舞いや生活･社会を構成する事象に追従して応答するのに十分なリアルタイム性を持つ.

（２）公開鍵暗号と PKI をベースとした暗号, 認証のメカニズムを有し, 社会のインフラを支えるユビキタス環境にふさわしい安全性と信頼性を実現できること.

（３） 情報家電やインターネットアプライアンスといった比較的乏しい計算機環境の上でも効率よく動作するように, 実行性能がよくかつ規模が小さいシステムになっていること.

（４） システムを管理するための労力が小さいこと. 具体的には, ユビキタスコンピューティング環境を構成する機器が設置されたら, たとえ停電等が起きても, 無設定で復旧し, 基本的に機器が故障するまで, メンテナンスする必要がない.

（５） 非専門家でも扱える簡便さを有すること. 例えば, 暗号・認証機構を知らない人でも, セキュアネットワーキングサービスを利用できること.

（６） 省リソース対応した回路技術を確立する. 特に, 省電力機能による電磁ノイズ発生の問題等を解決する.

（７） IP 網, デジタル方式の携帯電話網, PHS 網, 固定加入電話網, ADSL 網といった, 既存通信網との間のインターオペラビリティー機能を有すること.

（８） ユビキタスコンピューティング環境における典型的な協調･調停動作を複数実現することに成功し, その処理コードを分散透明な高い抽象度で記述できること.

■サブテーマ別の最終目標（詳細）

ア. 基盤通信システムの研究開発

（１） ユビキタスネットワーキングプロトコルのセッション層部分までの基本プロトコルの仕様を開発し, その正当性, 有効性を検証する.

（２） 上記のプロトコルを実現し, 評価を行う.

（３） ユビキタスネットワーキングの物理層･データリンク層を担う, Bluetooth や ISO 14443, IEEE 802.11, ISO 7816, 無線系電話プロトコルプロトコル等の中からネットワークシステムの上で

動作させるためのスタブ部分の仕様を開発すること.

（４） 既存のインターネット網である IP 網との間で相互運用と情報交換を可能にするゲートウェイ技術およびシステムを開発する.

（５） 基本機能として，認証機能，暗号機能を有すること.

（６） 認証・暗号機能の実現には，本研究のサブテーマ「カ.」で開発したセキュアハードウェアを十分に活用する.

（７） ソフトウェア規模は，十分小さい規模を想定する.

イ.ユビキタスコンピューティング環境を構成するシステム統合技術の研究開発

（１） IP 通信網や電話網などの既存の通信網との相互接続性を検証する.

ウ.超機能分散システム指向の開発環境の研究開発(ハードウェア部分)

（１） ユビキタスコンピューティング環境の構築の用いる標準開発プラットフォームとしてのハードウェアを開発する.

（２） その上で，サブテーマ「エ.」で開発した標準 OS が動作する.

（３） サブテーマ「カ.」で開発したデュアル型のセキュアチップを搭載している.

（４） サブテーマ「ア.（３）」で挙げた各種ネットワークプロトコルを搭載する.

（５） 音声 CODEC を備える.

（６） グラフィックチップを備える.

エ.超機能分散システム指向の開発環境の研究開発(ソフトウェア部分)

（１） 本研究サブテーマ「ア.」で開発するユビキタスネットワーキングプロトコルを標準機能で組み込み，それを本研究全体の標準プラットフォームとして利用する，標準リアルタイム OS を開発する.

(1-1) マルチタスク機能と，豊富なタスク間通信・同期機能を提供することができる.

(1-2) 省電力機能を有する.

(1-3) 本研究のサブテーマ「カ.」で開発したセキュアチップとの通信機能を有する.

（２） 本研究サブテーマ「ア.」で開発するユビキタスネットワーキン

グプロトコルに対して高抽象度のプログラミングインタフェースを提供するためのミドルウェアを開発する.

（３） 現実世界記述標準形式に関する研究開発

(3-1) ユビキタスコンピューティング環境が取り扱う実世界の各種環境情報情報の標準デジタル表現形式を策定する.

(3-2) ユビキタスコンピューティング環境が扱うあらゆるパラメータの表現を目指すため,その規模を例えると,「理科年表」のようなものになると考えている.

(3-3) 開発された標準記述形式は，ユビキタスネットワーキングプロトコルのプレゼンテーション層標準の一部として，全機器において使う.

(3-4) 表現の枠組みとしては,文字列形式である XML（eXtensible Markup Language）とバイナリ形式である TAD（TRON Application Databus）形式を構築する．特に後者は表現情報を効率よく表現できるための圧縮表現形式として，計算機資源が乏しいノードで利用する.

（４） 超機能分散プログラミングモデルに関する研究開発

(4-1) ユビキタスコンピューティング環境中に存在する莫大なノードの協調動作を高い抽象度でプログラミングできるプログラミングモデルおよび，そのプログラミング環境の開発を行う.

(4-2) ノード数は，数十から数万までを取り扱うことができ，ノードの分散性，動作の並列性をエンカプスレーションすることができる.

(4-3) あるユビキタスコンピューティング環境で動作していたソフトウェアをそのまま同じ機能をもった，他のユビキタスコンピューティング環境上でも稼動する移植性を有する.

オ．ユビキタスネットワーキングシステムのシステム工学的検証

（１） 本研究開発課題で構築したユビキタスコンピューティング環境の，一年程度の試験を行い，その期間の運用に耐えること.

（２） 現実社会における仕組みの中で運用しても，十分なセキュリティー強度，運用の容易性が達成できること.

（３） 情報通信分野の素人である一般ユーザでも十分本システムを使

いこなし，ユビキタスコンピューティング環境の機能の恩恵を受けられること．

（４） ユーザインタフェース部分には，ユニバーサルデザインが施されていること．

（５） 本研究開発課題で作成した実験レベルのシステムの運用データに基づき，それを都市レベルに拡大して普及させた場合の各種スケールファクターが確かめられたこと．

（６） 本システムを社会・生活の場に持ち込んでも，ユーザに不快感を与えたり，社会活動に悪影響を与えないこと．

カ．セキュアコンピューティングの基盤となるセキュアハードウェア

（１） コンタクトレス（無線）チャンネルのみを有するコンタクトレスチップと，コンタクトレス（無線）チャンネルとコンタクト（有線）チャンネルの双方を有するデュアルチップを開発する．

（２） コンタクトレス通信チャンネルの物理層・データリンク層のプロトコルは，ISO14443 Type-C 規格を満たす．

（３） コンタクト通信チャンネルの物理層・データリンク層のプロトコルは，ISO 7816 規格を満たす．

（４） 本課題で開発したユビキタスネットワーキングプロトコルで通信する機能を備える．

（５） PKI を使った公開鍵暗号技術に基いた暗号機能・認証機能を備える．

（６） 共通鍵暗号技術に基いた，実行効率のよい暗号機能・認証機能を備える．

（７） 耐タンパー性を有しており，悪意あるユーザからの不正操作から格納情報が守られる．

（８） ユビキタスコンピューティング環境を構成するノードに組み込むことで，そのノードの通信の安全性を向上できる．

キ．ユーザノードシステムの研究開発

（１） ユーザノードとは，ユビキタスコンピューティング環境の中で，利用者が直接接するユーザインタフェースをもった機器である．移動ノード，固定ノードとして，それぞれ複数種類のインテグレーションされたユーザノードを開発する．

（２） ユーザノードは，最終的にはサブテーマ「ウ．」で開発した標準ハードウェアを用い，サブテーマ「エ．」で開発した標準 OS，

ミドルウェアなどを利用して開発する.

（３） サブテーマ「ア.」で開発したユビキタスネットワーキングプロトコルを実現する.

ク. サーバノードシステムの研究開発

（１） サーバノードとは，ユビキタスコンピューティング環境を裏で支える基盤サーバ群を含む.

（２）サーバノードは，以下の機能を提供する.

- CA 局や鍵配布サーバを含む PKI（公開鍵インフラストラクチャ）機能
- 電子マネーや電子チケットの決済機能
- 価値情報の発行機能
- デジタルコンテンツの発行機能

（３） サーバノードも悪意ある攻撃から守るためにハードウェアに一定の耐タンパー性を持たせる.

3-2-2　中間目標

■全体を包括する最終目標

（１）公開鍵暗号と PKI をベースとした暗号，認証のメカニズムを有し，社会のインフラを支えるユビキタス環境にふさわしい安全性と信頼性を実現できること.

（２） 情報家電やインターネットアプライアンスといった比較的乏しい計算機環境の上でも効率よく動作するように，実行性能がよくかつ規模が小さいシステムになっていること. 基盤プロトコル全体で，200～300KB 程度のソフトウェア規模を狙う.

（３） 非専門家でも扱える簡便さを有すること.

（４） システムを管理するための労力が小さいこと.

ア. 基盤通信システムの研究開発

（１） ユビキタスネットワーキングプロトコルのセッション層部分までの基本プロトコルの仕様を開発する.

（２） 最終目標の記載欄で挙げ物理層・データリンク層プロトコルのうち，いくつかに関しては，そのスタブ部分の仕様開発を完了する.

（３） 基本機能として，認証機能，暗号機能を有すること.

（４） ソフトウェア規模は，十分小さいバイナリサイズを想定する.

イ.ユビキタスコンピューティング環境を構成するシステム統合技術の研究開発

（１） 相互接続性の検証が一部完了していること.

ウ.超機能分散システム指向の開発環境の研究開発(ハードウェア部分)

（１） ユビキタスコンピューティング環境の構築の用いる標準開発プラットフォームとしてのハードウェアを開発する.

（２） その上で，サブテーマ「エ.」で開発した標準 OS が動作する.

（３） サブテーマ「オ.」で開発したデュアル型のセキュアチップを搭載している.

（４） サブテーマ「ア.（３）」で挙げた各種 LAN，PAN のプロトコルを搭載可能である.

（５） 音声 CODEC を備える.

（６） グラフィックチップを備える.

エ.超機能分散システム指向の開発環境の研究開発(ソフトウェア部分)

（１） 本研究サブテーマ「ア.」で開発するユビキタスネットワーキングプロトコルを標準機能で組み込み，それを本研究全体の標準プラットフォームとして利用する，標準リアルタイム OS を開発する.

(1-1) マルチタスク機能と，豊富なタスク間通信・同期機能を提供することができる.

(1-2) 省電力機能を有する.

(1-3) 本研究のサブテーマ「カ.」で開発したセキュアチップとの通信機能を有する.

（２） 本研究サブテーマ「ア.」で開発するユビキタスネットワーキングプロトコルに対して高抽象度のプログラミングインタフェースを提供するためのミドルウェアを開発する.

（３） 現実世界記述標準形式に関する研究開発

(3-1) 内容的には，最終目標で記載したとおり．中間目標の時点では，標準形式の策定が完了している．それを実際のユビキタスコンピューティング環境に組み込んで実現するのは，

これ以後の年度に行うものとする.

（４） 超機能分散プログラミングモデルに関する研究開発

(4-1) ユビキタスコンピューティング環境中に存在する莫大なノードの協調動作を高い抽象度でプログラミングできるプログラミングモデルおよび，そのプログラミング環境の開発を行う．中間目標の時点で，その基本モデルは確立する．その実現や検証はそれ以後の年度に行うものとする.

オ．セキュアコンピューティングの基盤となるセキュアハードウェア

（１） 最終目標の挙げた中で，コンタクトレス（無線）チャンネルのみを有するコンタクトレスチップの開発が完了している.

（２） コンタクトレス通信チャンネルの物理層・データリンク層のプロトコルは，ISO14443 Type-C 方式である.

（３） この時点で開発された版のユビキタスネットワーキングプロトコルで通信する能力を有する.

（４） 共通鍵暗号技術に基いた，実行効率のよい暗号機能・認証機能を有すること.

（５） 耐タンパー性を有しており，悪意あるユーザからの不正操作から格納情報が守られること.

カ．ユーザノードシステムの研究開発

（１） ユーザノードとは，ユビキタスコンピューティング環境の中で，利用者が直接接するユーザインタフェースをもった機器である.移動ノード，固定ノードとして，それぞれ１種類以上のインテグレーションされたユーザノードを開発する.

（２） ユーザノードは，最終的にはサブテーマ「ウ.」で開発した標準ハードウェアを用い，サブテーマ「エ.」で開発した標準 OS，ミドルウェアなどを利用して開発する.

（３） サブテーマ「ア.」で開発したユビキタスネットワーキングプロトコルを備える.

キ．サーバノードシステムの研究開発

（１） サーバノードとは，ユビキタスコンピューティング環境を裏で支える基盤サーバ群を含む.

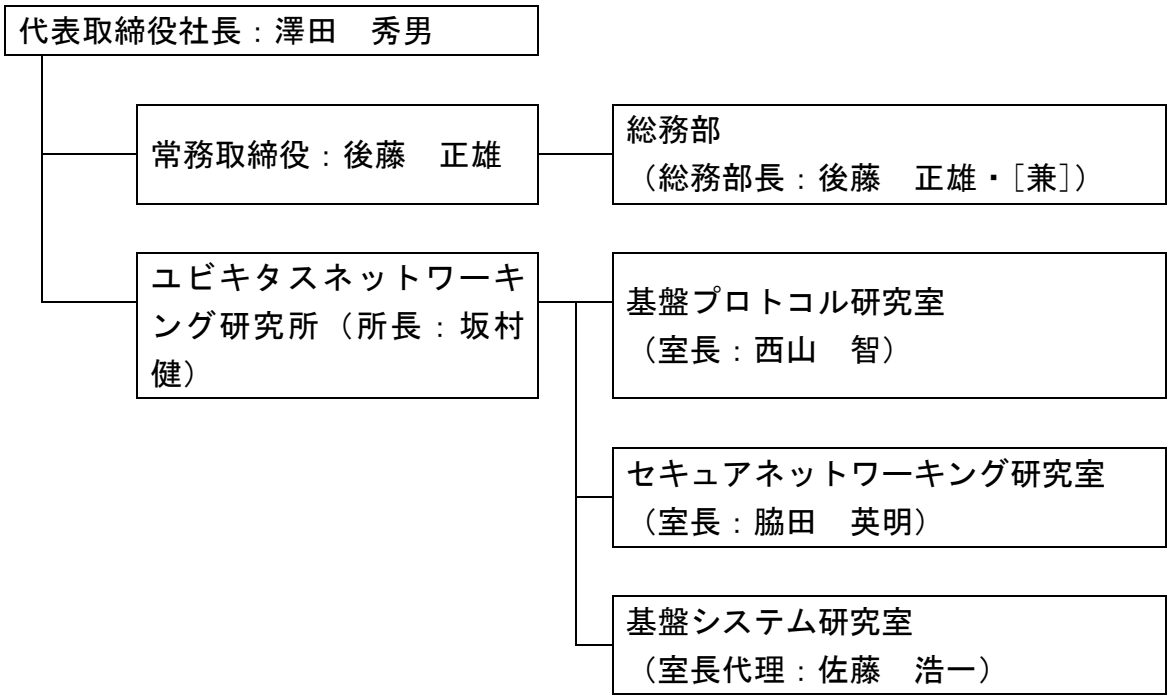
（２）中間目標の時点では，CA 局や鍵配布サーバを含む PKI（公開鍵イ

ンフラストラクチャ）機能の開発が完了している.

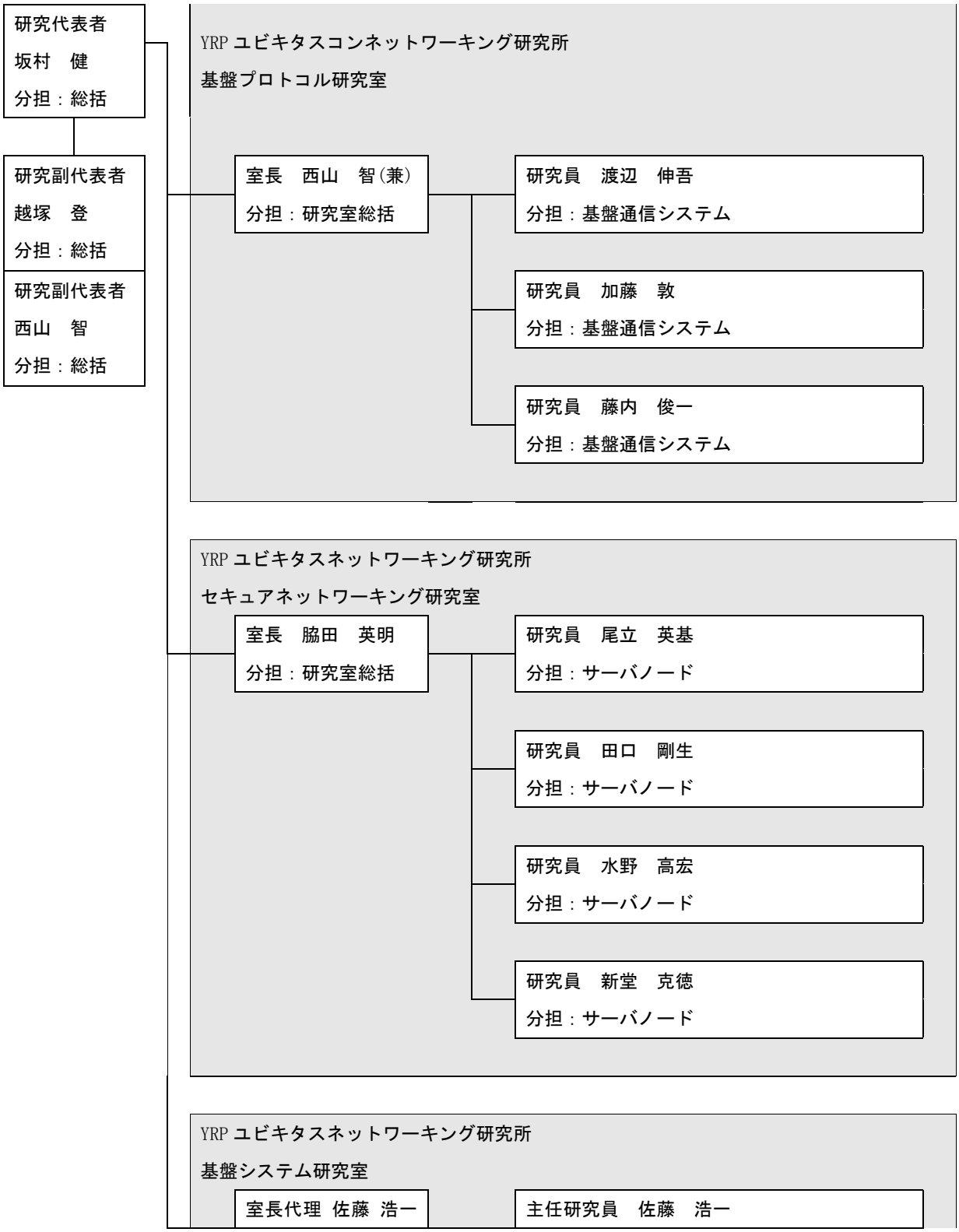
3-3 研究開発体制

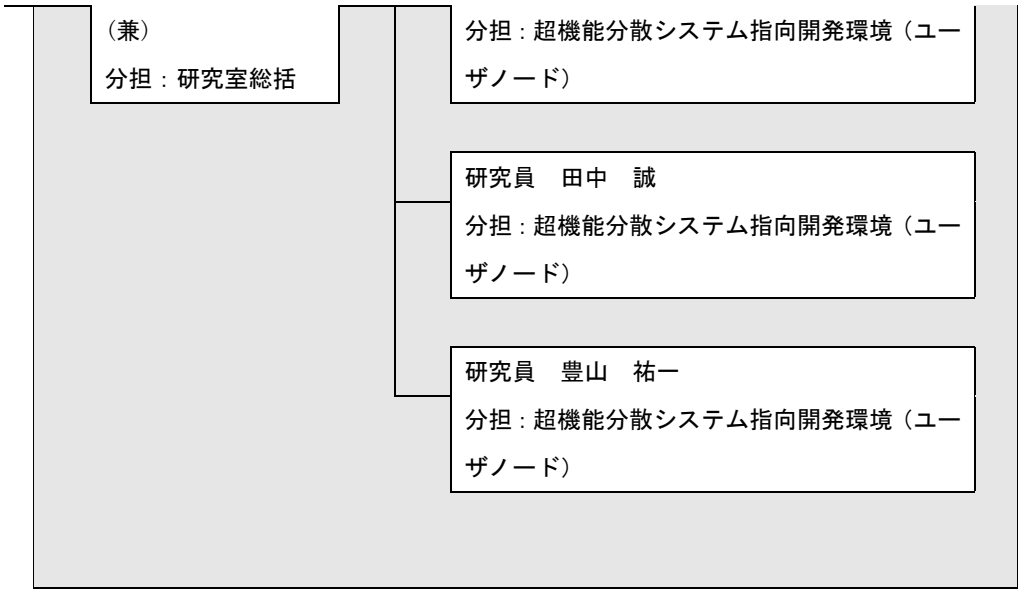
・研究開発管理体制

（株）横須賀テレコムリサーチパークにおける経理部門・研究部門の体制



・研究開発実施体制


研究代表者
坂村　健
分担：総括
研究副代表者
越塚　登
分担：総括
研究副代表者
西山　智
分担：総括
YRP ユビキタスコンネットワーキング研究所
基盤プロトコル研究室
室長　西山　智(兼)
分担：研究室総括
研究員　渡辺　伸吾
分担：基盤通信システム
研究員　加藤　敦
分担：基盤通信システム
研究員　藤内　俊一
分担：基盤通信システム
YRP ユビキタスネットワーキング研究所
セキュアネットワーキング研究室
室長　脇田　英明
分担：研究室総括
研究員　尾立　英基
分担：サーバノード
研究員　田口　剛生
分担：サーバノード
研究員　水野　高宏
分担：サーバノード
研究員　新堂　克徳
分担：サーバノード
YRP ユビキタスネットワーキング研究所
基盤システム研究室
室長代理 佐藤 浩一
主任研究員　佐藤　浩一


（兼）
分担：研究室総括
分担：超機能分散システム指向開発環境（ユーザノード）
研究員　田中　誠
分担：超機能分散システム指向開発環境（ユーザノード）
研究員　豊山　祐一
分担：超機能分散システム指向開発環境（ユーザノード）

第四章
研究開発の概要
（平成 15 年度まで）

4.1 研究開発実施計画

4.1.1 研究開発の実施計画(平成 14 年度)

4.1.1.1 研究開発の概要

平成 14 年度の研究開発の中心は、ユビキタスネットワーキング環境の基盤となる新しい基本方式の策定、およびその有効性を評価・検証するための、ハードウェア、ソフトウェアの実装である。その設計には、昨年度取り組んだ､ユビキタス型応用評価による知見が有効に活かされる。また、システムを試作する際には、やはり昨年度取り組んだ、研究開発基盤システムであるハードウェア､ソフトウェアを利用して構築する。

4.1.1.2 主要な研究題目と目標

本研究は、６つのサブテーマに分けて研究計画を立案している（全体計画の項目参照)。ここでは、その６つのサブテーマに沿って、今年度取り組むことを予定しているの主要な研究開発テーマについて述べる。

【サブテーマ１】
セキュアコンピューティングの基盤となるハードウェア

ユビキタスコンピューティング環境による夢を実現するためには､セキュリティーの確保が不可欠である。更に近年は、サイバーアタックやクラッキング、サイバーテロを想定した対策も求められている。ユビキタスコンピューティング環境を､ネットワーク経由のアタックから守るためには、セキュアな通信プロトコル、セキュアな通信システムの開発が不可欠である。そこで、本研究では、LSI 自体に不正アクセスが加えられないように加工を施した、いわゆる「耐タンパー性（Tamper Resistance)」を有するハードウェアを、ユビキタスコンピューティング向けチップとして新規に開発する。

その中でも本年度は、主にコンタクトレス（ノン・コンタクト）型の通信インタフェースを有するセキュアチップの研究を行う。現在の計画では、マルチメディア応用にも対応可能な（１）高性能なハイエンド型のセキュアチップと､身の回りのあらゆるものに埋込むことを想定した、安価で高信頼性の（２）超小型セキュアチップの研究を行う。

【サブテーマ２】

基盤通信システム

本研究ではユビキタス環境を構築するための、次世代通信プロトコルを研究開発する。特に本年度は、通信ノード・計算ノードの「偏在（ユビキタス）性」という特質部分を有効に扱うコア技術の研究開発を行う。まず第一にモバイル型の通信ノードが、様々な条件に応じて動的にネットワークを再構成する、（１）アドホックネットワーキングの研究である。第二に、ユビキタス環境を構成する据置型の通信ノードを対象とし、エンドユーザが自由に拡張することが可能な、（２）随意拡張型固定網通信プロトコルの研究である。また、通信プロトコルスタックの上位におけるプレゼンテーション層の研究として、ユビキタスネットワークプロトコルが扱う実世界情報のデジタル表現手法の研究として、（３）実世界データ研究・Everything ID 研究に取り組む。ユビキタス環境とは、実世界のあらゆる事象をコンピュータ上で処理し、デジタル通信で情報交換することと位置付けることができ、その際、実世界のあらゆる情報が、コンピュータの上で標準的な形で表現されることが不可欠であり、本年度は、その基幹部分のデータインフラの構築を行う。

【サブテーマ３】
ユーザノードシステム

ユビキタスコンピューティング環境を構成するノードの中で、ユーザと直接接することが想定される機器類の研究開発である。本サブテーマでは、ユビキタスコンピューティング環境での利用条件に適合したユーザノードの構成方法・機構を研究開発する。特に、本年度は、（１）カームネットワーキング／カームコンピューティング、（２）強化現実型ユーザインタフェース、（３）位置検出機構の３テーマを中心的に扱うことを計画している。

（１）カームネットワーキング／カームコンピューティング

ユビキタスコンピューティング環境では、あらゆる場所に膨大な数のネットワークノード・コンピューティングノードが設置される。これらは、電力などの資源を浪費することなく、最小限の資源利用で効率的に動作することが不可欠である。そこで、こおうした無数にちりばめられるコンピュータを想定し、ハードウェアおよび OS や BIOS レベルの基盤ソフトウェアにおける、新しい電源制御機構を含む省電力機構の実現技術を研究開発する。

（２）強化現実型ユーザインタフェース

ユビキタス環境のユーザノードに対して、高い対ユーザ親和性（User Friendliness）を実現するためには、コンピュータの存在を意識することなく、自然な対話が行えることが重要である。そうしたユビキタス環境におけるユーザとのインタフェースをつかさどる技術として、本年度は、強化現実型ユーザインタフェース（Augmented Reality）の研究開発を進める。

（３）位置検出機構

上記の強化現実型ユーザインタフェースの特徴は、コンピュータやネットワークによって構成される電子的な仮想空間と現実空間を融合することによって、現実世界において、より高いユーザ親和性と IT サービスを提供する点にある。これを実現するために、最も重要な要素技術の一つで、現在まだ実用レベルに達していないものに、位置検出メカニズムがある。本年度は、特に屋内、つまり GPS などのグローバルな位置検出インフラが利用できない実環境、における高い精度のユーザ位置検出技術の研究開発を行う。

【サブテーマ４】
サーバノードシステム

サーバノードシステムは、ユビキタス環境をバックエンドで支える、大量の計算資源と通信資源を有するインフラシステムである。サーバノードシステムにはさまざまな役割があるが、今年度は特に、セキュリティー基盤としてのサーバーノードシステムの研究に注力する。最初のサブテーマにある、セキュアハードウェアの研究開発と連携し、そこで研究開発されるセキュアチップのセキュア基盤のバックエンドとして動作するものを研究開発する。

本年度のセキュア基盤の研究の主要な課題は、セキュリティーを実現する機構をコンパクトに実装すること（（１）ユビキタス PKI）と、公開鍵暗号系の認証処理を行うときの PKI の応答性を向上させる（２）リアルタイム PKI 技術の確立である。

【サブテーマ５】
システム統合技術

ユビキタス環境において、セキュリティーをエフォートレスに実現す

る手法が求められている。そこで、本年度の研究では、ユーザの身体的・生理的特性を利用したバイオメトリクス型の認証機構を、ユビキタス環境に適用するために大規模化する研究に取り組む。そのためには、各ユーザ端末にバイオメトリクス検知用のセンサーを格納し、認証情報はサーバシステム上でセキュアに集中管理し、認証情報を通信回線で交換する、分散型バイオメトリクス認証処理である。この機構の実現には、セキュアハードウェアの研究、ユーザ端末の研究、サーバシステムの研究と連携をとった、統合的な技術として研究開発を進める。

【サブテーマ６】
超機能分散システム指向開発環境の整備

昨年度から継続し、ユビキタスコンピューティング環境の基盤システムとなる、ハードウェア（MPU、基本ボード、等）、ソフトウェア（オペレーティングシステム、各種ミドルウェア、等）の研究開発を継続する。特に本年度注力するのは、以下のテーマである。

（１）ワンボード型エンベディドアーキテクチャ
現在の PC に匹敵する計算資源や通信資源を有するハイエンドクラスのユビキタス環境向けアーキテクチャの研究である。実際に、本アーキテクチャに基づいたシステムを試作して、評価する。

（２）ワンチップ型エンベディドアーキテクチャ
最も小型で、安価なユビキタス環境向けの計算機ノードアーキテクチャを研究する。この場合、通信インタフェースをも含んだ、全システムが、ワンチップ上に実装されることを想定している。

（３）ミドルウェア研究
上記アーキテクチャシステムをターゲットとした、ミドルウェアシステムの研究開発を行う。主に、分散ネットワーキングシステムのミドルウェアや、暗号・認証などのセキュリティーシステムのミドルウェアの研究開発を行い、高い品質のユビキタス環境ソフトウェアを高い効率で開発するための手法を研究する。

4.1.2 研究開発の実施計画（平成 14 年度）
4.1.2.1 研究開発の概要

平成 15 年度の研究開発の中心は、昨年度取り組んだユビキタスネットワーキング環境の基盤となる基本方式の更なる発展である。

昨年度、実装・評価・検証したハードウェア、ソフトウェアをベースに、その機能、性能、および信頼性の向上を目指す。また、ユビキタス型応用に必要な技術要件を更に高い水準で満たすことのできる方式研究およびハードウェア、ソフトウェアの実装を行う。

4.1.2.1 主要な研究題目と目標

本研究は、６つのサブテーマに分けて研究計画を立案している（全体計画の項目参照)。ここでは、その６つのサブテーマに沿って、今年度取り組むことを予定している主要な研究開発テーマについて述べる。

【サブテーマ１】
セキュアコンピューティングの基盤となるハードウェア

ユビキタスコンピューティング環境による夢を実現するためには、セキュリティーの確保が不可欠である。更に近年は、サイバーアタックやクラッキング、サイバーテロを想定した対策も求められている。ユビキタスコンピューティング環境を、ネットワーク経由のアタックから守るためには、セキュアな通信プロトコル、セキュアな通信システムの開発が不可欠である。そこで、本研究では、LSI 自体に不正アクセスが加えられないように加工を施した、いわゆる「耐タンパー性（Tamper Resistance)」を有するハードウェアを、ユビキタスコンピューティング向けチップとして新規に開発する。

本年度は、昨年度に引き続き、高性能なコンタクトレス(無線)チャンネルを有し、16bit のワード幅を持つ CPU を搭載した（１）ハイエンド型のセキュアチップを研究開発する。また、更なる暗号処理高速化技術、生体認証との連携技術、電子価値を安全に管理／送受できる技術についての研究を実施する。安価で高信頼性の（２）超小型セキュアチップについては、実フィールドにおける耐環境性の向上技術について研究する。

【サブテーマ２】
基盤通信システム

本研究ではユビキタス環境を構築するための、次世代通信プロトコルを研究開発する。特に本年度は、セッション層までの次世代通信プロトコルを確立するために、以下の研究を行う。まず第一にモバイル型の通信ノードを主な対象として、ユビキタスアプリケーションや利用可能な通信環境に応じて最適な通信経路を複数の通信メディアから動的に選択してユビキタス通信を行う、（１）シームレス通信の研究である。これに必要となる、ネットワーク層からセッション層にいたる次世代通信プロトコルの研究開発を行う。第二に、ユビキタス環境に遍在する情報機器、特にセンサ等の超小型機器が多数存在する場合において効率的に通信を行うための、（２）ユビキタス情報提供・制御用プロトコルの研究である。このために、ネットワーク層以下の次世代通信プロトコルの研究を行う。また、通信プロトコルスタックの上位におけるプレゼンテーション層の研究として、ユビキタスネットワークプロトコルが扱う実世界情報のデジタル表現手法の研究として、（３）実世界データ研究・Everything ID 研究に引き続き取り組む。具体的には、実世界のあらゆる情報が、コンピュータの上で標準的な形で表現されるための符号化方式や番号体系（いわゆるユビキタス ID）の研究開発を行う。

【サブテーマ３】
ユーザノードシステム

ユビキタスコンピューティング環境を構成するノードの中で、ユーザと直接接することが想定される機器類の研究開発である。本サブテーマでは、ユビキタスコンピューティング環境での利用条件に適合したユーザノードの構成方法・機構を研究開発する。ユビキタスコンピューティングにおいてユーザと直接対話するアプリケーションの特徴として、コンテクストアウェア性がある。そこで本年度は特に、位置や環境、利用者等コンテクストに応じたユビキタスコンピューティング実現のために、以下の技術の研究開発を行う。

（１）コンテクスト情報記述・管理方式

一般にコンテクストとは、位置や時刻、温度等の環境から得られる情報や、ユーザに関する情報（例えば個人属性、嗜好、行動履歴など）などの総称である。これらの情報を記述し、効率的に管理・提供するための方式について研究開発する。

（２）位置検出機構

一般にコンテクストとして、位置が最も有力な特性である。今年度は、複数の位置検出手段及び対象オブジェクトに関するセマンティクスを総合的に利用して、位置を検出する手法について研究開発を行う。

【サブテーマ４】
サーバノードシステム

サーバノードシステムは、ユビキタス環境をバックエンドで支える、大量の計算資源と通信資源を有するインフラシステムである。今年度は、昨年度研究を行ったユビキタスＰＫＩを大量かつ高速に動作させるための（１）ユビキタスＰＫＩサーバの研究、およびユビキタス基盤サーバとして、ノードが異なるネットワークに存在しても位置を知ることができる（２）アドレス解決サーバ、電子的な価値情報をセキュアに発行することができる（３）セキュア発行サーバの研究開発を実施する。更にあらゆるものにチップが搭載された環境で個々を認識する手段として、現実世界記述標準形式を用いたユビキタスＩＤをバックエンドで支える（４）ユビキタスＩＤサーバの研究開発を進める。

【サブテーマ５】
システム統合技術

ユビキタス環境においては、次世代通信プロトコルを具備するユビキタスネットワークと、既存のネットワークとの相互接続機能が重要となる。そこで、本年度の研究ではこれまで開発した次世代通信プロトコルと既存ネットワークプロトコルとの相互接続技術を研究する。具体的にはネットワーク層からミドルウェア層でのアドレス変換技術や情報配信技術により、総合的に最適なシステム統合方式が選択できる基盤を研究開発する。

【サブテーマ６】
超機能分散システム指向開発環境の整備

昨年度から継続し、ユビキタスコンピューティング環境の基盤システムとなる、ハードウェア(MPU、基本ボード、等)、ソフトウェア(オペレーティングシステム、各種ミドルウェア、等)の研究開発を継続する。特に本年度注力するのは、以下のテーマである。

（1）ユーザノードシステム

昨年度研究を行ったワンボード型エンベディッドアーキテクチャおよびミドルウェアを応用して、ユーザノードシステムの研究開発を行う。ユーザノードとは、ユビキタスコンピューティング環境の中で、利用者が直接接するユーザインタフェースをもった機器である。

（2）ミドルウェア

高抽象度のプログラミングインタフェースを提供するミドルウェアとして、ユビキタスコンピューティングに適した軽く規模の小さい Java 言語処理システムを開発する。

4.2 研究開発の実施内容（平成 15 年度まで）

委託業務の内容は，６つのサブテーマに分けて実施した．

【サブテーマ１】セキュアコンピューティングの基盤となるハードウェア

【サブテーマ２】基盤通信システム

【サブテーマ３】ユーザノードシステム

【サブテーマ４】サーバノードシステム

【サブテーマ５】システム統合技術

【サブテーマ６】超機能分散システム指向開発環境の整備

以下，それぞれのサブテーマについて，委託業務の実施内容について述べる．

【サブテーマ１】
セキュアコンピューティングの基盤となるハードウェア

（1）高性能なハイエンド型のセキュアチップ

- ハイエンド型セキュアチップの最初のバージョンとして、8 ビット CPU を用いたセキュアデータキャリアチップの研究開発を行った。
- チップには、セキュリティーが強化されたセキュアプロトコル(eTP)を実装した。本実装では、暗号関数を５種類サポートし、従来技術と比較して強固なセキュリティレベルを実現した。
- ハイエンド型セキュアチップの改良バージョンとして、16 ビット CPU を用いたセキュアデータキャリアチップの研究開発を行

った。

- 多様な環境下での利用が想定されるセキュリティチップにアクセスするためのクライアントライブラリの設計・実装を行った。多様な環境として，セキュリティチップへの物理的なアクセス方法，チップとの通信および認証の方式，およびチップとの通信プロトコルの 3 点を想定し，これらを抽象化した。そのうえで，これらの環境をアプリケーション側が自由に組み合わせて使えるようにクライアントライブラリの設計・実装を行った。
- ユビキタスコンピューティング環境での利用、および生体情報の外部接続型セキュリティデバイスでの管理を想定した、ユビキタス端末向け生体認証機能に関して、国際標準規格 ISO 15408 に準拠したプロテクションプロファイルの開発を実施した。

（２）超小型セキュアチップ

チップサイズが 1mm 以下の超小型セキュアチップの技術調査および基本仕様の検討を実施した。

またパッシブ型の RFID の通信状況の環境依存性に関して、水分を多く含む食品や薬剤、金属を含むパッケージ類に関して、実際に RFID をこれらのモノに貼り付け、通信試験を行った。

【サブテーマ２】基盤通信システム

（１）シームレス通信の研究

- ユビキタスネットワーキングプロトコルの研究の一環として、ユーザノードの移動時にも可能な限り通信を提供し、また複数の通信手段が利用可能な場合により適した通信手段により通信するという、シームレス通信の研究を行った。具体的には、IETF で標準化されているネットワークプロトコルであるモバイル IP をベースに、応用の要求に応じて通信手段を使い分ける通信手段使い分け方式と、提案方式に対応するルータと通常のルータが混在する環境において移動に伴う通信不可時間を短縮する高速・高信頼化方式を考案した。
- 考案方式について情報処理学会第 66 回全国大会に、それぞれ「モバイル端末における応用の要求に応じた通信メディアの使い分けの提案」、「既存ルータ混在環境におけるモバイル IP ハンドオーバの高速・高信頼化の提案」と題して発表した。

（２） ユビキタス情報提供・制御用プロトコルの研究

- データリンク層

  データリンク層のプロトコルには既存のトークンパッシング方式を改良したプロトコルを開発して実装した。通信形態は、ユニキャストとマルチキャスト（ブロードキャストを含む）をサポートした。また、パケットヘッダの特定のフィールドを検査することによって、優先制御を行うリアルタイム通信と、ベストエフォート通信の処理ができるようにした。ネットワーク上の論理アドレスについては、機器同士が協調してアドレスを管理できるアルゴリズムを開発した。機器が削除された場合でも、論理的に近くにいる機器がタイムアウトすることによって削除されたことを検知するアルゴリズムを開発した。また、本レイヤを LSI 化することでシステム全体の処理速度の向上を図った。

- トランスポート層

  トランスポート層のプロトコルとしては、コネクション型とコネクションレス型のプロトコルを用意した。確認応答を有し、信頼性の高い通信を可能とするプロトコルも用意した。さらに応答確認のタイミングを送信側が指定できるようにすることで、ネットワークに負荷をかけずに信頼性を確保することも可能となった。

- セッション層

  セキュリティや認証のための機能を提供した。相互認証と同時に鍵交換を行い、セッション中は暗号通信が行なわれるセキュアセッション機能、リアルタイム応答が要求される場合のライトウェイトセッション機能を備えた。また、階層構造をもつネットワーク構成においてセキュリティの強度を上げるために、各ドメイン単位で異なる鍵を使用するなどの配慮も行った。なお、目標設定時はセッション層でセキュリティや認証機能を実現する予定であったが、セキュリティの強度を上げるため、これらの処理はネットワーク層の上層に実装した。

（３） 実世界データ研究・Everything ID 研究

- 実世界のあらゆる情報をコンピュータ上で標準的な形式で記述するために必要となる番号体系(ユビキタス ID)を規定し、その番号体系に基づき情報を取得するための番号解析プロトコルを開発した。

【サブテーマ３】ユーザノードシステム

（１）カームネットワーキング／カームコンピューティング

ハードウェアおよび OS や BIOS レベルの基盤ソフトウェアにおける、新しい電源制御機構を含む省電力機構のためのプロトコルと制御方式を開発した。

（２）強化現実型ユーザインタフェース

下記に述べる超音波センサと RFID を組み合わせた位置検出機構の研究及び、【サブテーマ２】で実施した屋内用センサネットプロトコルで得られる位置情報やユビキタス環境情報を実際の屋内モデルにマッピングしてユーザに提示する研究を実施した。

（３）位置検出機構

現実世界において、より高いユーザ親和性と IT サービスを提供するために、特に屋内、つまり GPS などのグローバルな位置検出インフラが利用できない実環境、における高い精度のユーザ位置検出技術として、超音波センサと RFID を組み合わせた位置検出と、画像認識による位置検出方式を開発した。また取得した位置を管理するためのサーバの開発を行った。

また、平成 15 年度は複数のソースを利用した位置検出メカニズムとして、赤外線 ID タグと、RFID・非接触 IC カードの RF を利用した位置検出機構を研究し試作した。

（４）ユーザノードシステム

本研究全体の標準プラットフォームのアーキテクチャを基に、本研究で開発された標準リアルタイムＯＳおよびミドルウェアを利用して、ユビキタスコンピューティング環境において利用者が直接接するユーザノードを、移動ノード、固定ノードについて、それぞれ開発を完了した。移動ノードに関しては、ユビキタスコミュニケータ（UC）と UC-Phone の二種類を開発した。（なお、この UC の方は、ユーザノードの開発環境としても利用した）

（５）コンテキスト情報記述・管理方式

コンテキスト情報の記述・管理方式として、当研究所では、U-TAD（Ubiquitous TAD）というフレームワークの構築をしているが、今年度

は特に、位置情報に関する記述方法を中心的に研究した。

【サブテーマ４】サーバノードシステム

（１）ユビキタス PKI

従来の PKI 証明書(X.509)と比較し、リソースの少ないユビキタス機器で利用できる証明書の研究開発を実施した。具体的には、ユビキタス環境での情報セキュリティを保証する証明書の必要項目を抽出し、暗号方式の異なる複数種の暗号鍵を円滑に取り扱う、独自の形式を検討した。

（２）リアルタイム PKI

【サブテーマ１】で実装した eTP との組み合わせで、公開鍵暗号系の処理を IC カードでリアルタイムに実現できる認証方式を研究開発した。

上記のプロトコルをサーバノードシステムに実装し、ユビキタス PKI の基本アーキテクチャを構築した。

（３）アドレス解決サーバ

セキュア IC チップに付与される ID によるアドレス解決を実現するため、セキュア IC チップに付与される ID と IP アドレスの組を管理する ID 解決サーバを開発した。

（４）セキュア発行サーバ

セキュア基盤ネットワークで流通する電子的価値情報を発行するセキュア発行サーバを開発した。

（５）ユビキタス ID サーバ

ユビキタス ID とその ID か付与されたオブジェクトに関する情報を提供するサーバの名前やアドレスの対応関係を管理し、ユビキタス ID 解決プロトコルに従った手順でユーザノード等に情報提供するユビキタス ID サーバを開発した。

【サブテーマ５】システム統合技術

（１）分散型バイオメトリクス認証処理

分散型バイオメトリクス認証処理に必要な要素技術の研究開発を実施した。

- 基本処理方式の検討を行った。
- 必要なハードウェアを試作した。
- デバイスドライバ等、基本的なソフトウェアの開発をおこなった。

（２）次世代通信プロトコルと既存ネットワークプロトコルとの相互接

続技術

総合的に最適なシステム統合方式が選択可能な基盤を実現するために、ユビキタス環境におけるソフトウェア等のコンテンツの流通を促進、ライセンス制御可能なコンテンツ配布流通用ソフトウェアを設計し、その一部を試作した。

【サブテーマ６】超機能分散システム指向開発環境の整備

（１）ワンボード型エンベディッドアーキテクチャ

ユーザノードの開発環境として、昨年度までに研究開発した、U-Card のアーキテクチャをベースとして、それに RFID インタフェースなどを付与したユビキタスコミュニケータ（UC）を研究開発した。

ユビキタスコンピューティングに関連する各種デモンストレーションの開発に応用することで、その有用性を確認した。具体的には、RF-ID タグと組み合わせてグラフィックユーザインタフェースにモノの情報を有機的に表示するシステムや、電子的な価値情報を転々させるシステムを試作完成させ、評価した。

（２）ワンチップ型エンベディッドアーキテクチャ

５００円玉サイズのシステムとして、仕様策定および開発が完了した。セキュアチップを接続するインタフェースおよび各種通信機能を搭載したシステムとなっている。

（３）ミドルウェア研究

次のミドルウェアを開発した。

- 主なネットワークプロトコルを実現するミドルウェア
- 非接触 IC カードとセキュアな通信を行うための暗号通信ミドルウェア
- サーバと協調して認証を行うミドルウェア
- J2ME CDC

  高抽象度のプログラミングインタフェースを提供するためのミドルウェアとして Java 言語環境を開発した。ユビキタスコンピューティングに適する、軽くソフトウェア規模の小さい Java Virtual Machine および多漢字プロファイルの構築を行った。また、サーバサイドでは、研究所にて開発されたユビキタスネットワーキングプロトコルライブラリ群を Java 環境上で提供するために必要なクラスライブラリの構築を行った。

第五章
研究開発実施状況
（平成 15 年度）

5-1 セキュアコンピューティングの基盤となるハードウェアの研究開発

5-1-1 高性能なハイエンド型のセキュアチップ

ハイエンドセキュアチップのハイエンド版として 16 ビット CPU を用いたセキュアデータキャリアチップ(以下 16 ビット型セキュアデータキャリアチップと呼ぶ)の研究開発を行った。以下では 16 ビット型セキュアデータキャリアチップの詳細を示す。

(ア) 16 ビット型セキュアデータキャリアチップ命令の概要

16 ビット型セキュアデータキャリアチップには、認証通信を開始して、暗号化したデータ通信をおこなう機能が基本としてそなえられている。

16 ビット型セキュアデータキャリアチップ における認証は、以前の 8 ビット CPU を利用したチップとはことなり、公開鍵暗号技術 (PKI)をもとにしている。個々の 16 ビット型セキュアデータキャリアチップには、個々に割り付けられた公開鍵と秘密鍵のペアーが格納されている。また、公開鍵にたいしては証明書発行局 (CA)から発行された署名つきの証明書が格納されている。

認証通信の初期の段階においては、個々の 16 ビット型セキュアデータキャリアチップは通信相手の 16 ビット型セキュアデータキャリアチップの ID と、その ID に対応する公開鍵に対応する秘密鍵を通信相手がもっていることを PKI 演算を利用することで確認する。確認できないときには通信をおこなわない。

なおいろいろな認証通信をサポートする枠組を提供するということで、この認証仮定は 2 段階のハンドシェイクを行なうようにつくられて 2 番目のコマンドの応答を得て初めて認証が成功したかどうかがわかるという前提のものになっている。

相手が公開された ID と証明書中にはいった公開鍵に対応する秘密鍵をもっていることを確認したあとで、対向する 16 ビット型セキュアデータキャリアチップとの一時的な通信セッションに使う共通鍵を生成する。

以降の通信は、この共通鍵を使い暗号化され、さらに外部からの干渉を検出するために、HMAC (Hashed Message Authentication Code) を付加することで通信経路に流される。これにより 16 ビット型セキュアデータキャリアチップ間

の End-to-End 通信をセキュアなものにすることができる。共通鍵を使った暗号アルゴリズムとしては AES (Rijndael)など複数のアルゴリズムを試験実装した。

なお、アプリケーションプログラムないしは、人間ユーザからコマンドを 16 ビット型セキュアデータキャリアチップに送信するための共通鍵を使った認証パスも用意されている。(これは直接チップがささった機器でのみの利用が想定されている。遠方の通信機器の間の 16 ビット型セキュアデータキャリアチップの通信は上記の PKI を利用した認証通信を利用することが想定されている。) この部分の暗号アルゴリズムは以前のチップのアルゴリズムを踏襲している。

このように設立した認証暗号通信経路で 16 ビット型セキュアデータキャリアチップにコマンドを送出し、その結果として各種処理をおこない、結果をうけるというのが 16 ビット型セキュアデータキャリアチップの動作の基本である。

16 ビット型セキュアデータキャリアチップのデータ処理のアーキテクチャは以下のようなものとなる。

内部に「フォルダー」が作られる。データを格納するための 「ファイル」を「フォルダー」中に複数個作成できる。ファイル中の「レコード」にたいしてデータを読み書き機能も用意されている。

ただし、試験的に作成したチップではフォルダーは 1 レベルで、トップレベルだけに存在して、レコードはファイルの中に一個だけ存在する。

ファイルに対する操作は以下のようなものである。

- 作成、削除。
- レコードの読み書き。
- ファイルの属性の設定、読みだし。

属性とはファイルの読み書きなどの操作に対する許可の設定である。

これらのファイルの内容を、別に与える鍵を使い共通鍵アルゴリズムを使って暗号化、復号し、さらに暗号、復号のステップ数に応じて課金をおこなうためのプリミティブも用意されている。このための命令としては以下のようなものが用意されている。

- 課金情報をもつ共通鍵の設定と削除。
- 上記の共通鍵をつかった課金処理をともなう暗号と復号操作。

これらの暗号、復号の対象となるデータサイズを大きくすることが今後の課題である。

ファイルの作成、削除、レコードの書き換えについては、いくつかの連続す

る操作が データベース的に ATOMIC (原子的) におこなえるように 「トランザクション」処理をおこなうためのセッションの開始が指定できるようになっている。一連の操作が成功したときに「コミット」することで、これらの操作の結果が永久的にチップ内部のメモリに保存される。「アボート」することで、これらの一連の操作の結果は無視されて、トランザクション開始以前の状態がそのままチップ内部にのこる。

なお、ファイルをひとつの 16 ビット型セキュアデータキャリアチップから 別の 16 ビット型セキュアデータキャリアチップに外部からコマンドを送ることなく、送信元の 16 ビット型セキュアデータキャリアチップが自動的にコマンドを別の 16 ビット型セキュアデータキャリアチップに発行することで行なえる。これがファイル転送の機能で転々流通の基礎となるものである。

PKI を利用することで、秘密鍵、公開鍵、ならびに公開鍵に対する証明書を書き込む必要が生じた。このために、これらの認証に使う暗号関連データの初期設定を行なうための命令ももうけた。実験目的で、暗号鍵ペアは外部から与えることもできるし、内部でセキュアに発生することもできる。これらの初期化命令群は、実際にチップを作成する半導体工場、ならびにチップを IC カード形状に作成する IC カード会社の処理に依存する場合が多いとおもわれる。16 ビット型セキュアデータキャリアチップの場合には、これらの依存する命令群を抽象化して汎用性をもたせた命令群を提供することで、多くの会社の実装に対応できることを目標としている。この目的を達成しているかを実証するのが今後の実験の調査目的のひとつとなる。

なお、PKI の利用にともない、証明書の発行が必要となるが、これの目的のプログラムを複数作り、それぞれことなる証明書発行局の鍵を利用して、証明書の発行、証明書の失効操作を行ない、16 ビット型セキュアデータキャリアチップの間で問題なく認証暗号通信が行なえることを広範な実験で実証されている。

さらに、16 ビット型セキュアデータキャリアチップ では認証通信に使う内部 PKI 計算機能を、外部機器の VPN (Virtual Private Network) の鍵計算に使うための機能を提供する。これについては ユビキタス PKI 機能として外部機器でも使うので、以下に節をあらためて説明する。

(イ) VPN 機能

16 ビット型セキュアデータキャリアチップ の VPN のサポート機能は、以下

のような枠組みを想定している。

VPN は一時的な共有鍵を、公開鍵暗号を使いなんらかの方法で生成するものとする。VPN をサポートする対象（VPN に参加するノード、それぞれのノードの公開鍵に対する証明書を発行する証明書発行局 (CA)、それらの証明書の失効を処理する CRL の管理などをまとめて VPN ドメインと呼ぶ。

16 ビット型セキュアデータキャリアチップ では VPN ドメインは複数個サポートできるようにする。

また、これらの VPN ドメインの使う CA、CRL 管理サーバーは 16 ビット型セキュアデータキャリアチップ本来のセキュアな公開鍵による認証通信につかう CA などとは異なるものとする。

このような設計方針の前提で、16 ビット型セキュアデータキャリアチップ には以下のような命令を用意した。なお、PKI 暗号アルゴリズムと指定は DH 鍵交換に多少修正を加えたものを想定している。

- 16 ビット型セキュアデータキャリアチップ 内部で VPN 用 PKI に使う 公開鍵、秘密鍵ペアーをつくり、あとで利用できるようにする。
- 16 ビット型セキュアデータキャリアチップ 外部から VPN 用 PKI に使う公開鍵、秘密鍵ペアーを与えて設定する。つくる。
- 16 ビット型セキュアデータキャリアチップ 内部にある VPN 用公開鍵の証明書を読出す。
- 16 ビット型セキュアデータキャリアチップ 内部の VPN 用公開鍵証明書の更新を行う。
- 16 ビット型セキュアデータキャリアチップ 内部の VPN 用公開鍵を読出す。
- 相手ノードからおくられてきた VPN 用証明書の妥当性チェックと、VPN ドメインに参加するアクセス権利のチェック。
- 16 ビット型セキュアデータキャリアチップ 内部で、VPN 用一時共有鍵の生成の計算につかう乱数を生成する。
- 16 ビット型セキュアデータキャリアチップ 内部で生成された VPN 用一時共有鍵を読出す。

この VPN 機能を利用したユビキタス PKI 機能については、後述する。

（ウ）16 ビット型セキュアデータキャリアチップにおける改良点

(1) メディアに依存するパケットフォーマットの適正化

16 ビット型セキュアデータキャリアチップは、本来直接対向するリーダー、ライター装置との通信しかできず、ネットワーク対応はそれらの対向する装置でのソフトウェアに依頼してコマンドパケット、ならびに応答パケットを転送してもらうことに依存している。

従来、ad hoc に決められたデータフォーマットでの通信をおこなっていたが、各種メディア（TCP/IP, ISO 7817, ISO 14443, USB などのメディア）での適切なパケット転送フォーマットを考案してそれによる実験システムに移行を考慮している。

TCP/IP の接続の場合には、リーダー／ライターデバイス上に存在して外部遠方サーバーとの通信を司るプロキシ機能をもつプログラムとサーバーの間での TCP ソケットの削除に対するタイミングの理解がアプリケーション毎に異なるために問題が発生した。

この問題は大規模な実験を行ない、複数のソフトウェア作成者がつくったサーバーと相互に通信を行なうことで初めて明らかになった。このような誤解を防ぐためのソフトウェア通信上の規定も設けた。

(2) チップサポートライブラリの複雑化の回避

今後各種のセキュアチップを開発する可能性もあり、サポートソフトウェアの無用な複雑化をさけるために、ソフトウェアの環境の大幅な作り替え作業をおこなった。

過去に作ったチップと最新の比較を考えると大きな変化は以下のようになる。

旧 8 ビットチップ - 非ネットワーク対応
トランザクション なし。
PKI は利用しない。共有鍵暗号のみ。

新 16 ビットチップ - ネットワーク対応
トランザクションあり。
ファイルの譲渡をチップが自律して行なえる。
PKI を積極的に利用する。

この機能の違いと将来における多品種チップの開発の可能性を考慮して、チップの種類および使用環境をあまり意識せずにアプリケーションからチップ機能を利用するためのクライアントライブラリを作成することにして、設計と実装をおこなった。（詳細は後述）

(3) その他の細かな変更点

上でのべたような全体の取組以外に、実際に 16 ビット型セキュアデータキャリアチップを利用して実験システムを運用するうちに発見された問題点などを本年の研究では解決した。

- DES のみのコマンドパケット､応答パケットの暗号機能はセキュリティの点で弱すぎるので、実験的な意味は別として、将来の利用からは削除する方向でライブラリなどを整備している。一方で、トリプル DES は実行で 100 ビット前後のセキュリティ強度があるので、そのまま残した。
- 暗号処理の中に不適切な 0 パディングを行なう部分がみつかったものを乱数でパディング処理するように変更してみた。
- 接触インターフェイスは、実験を短期間に行なうために、かならずしも ISO 7816 の物理層に完全準拠でない実装をおっていたが、これを完全に準拠するバージョンも試作した。ただし、これによりソフトウェアドライバの変更も必要となった。非常に残念なことに、このように変更したことで市販のリーダー／ライター装置が容易に使えるようになることが期待されたが、16 ビット型セキュアデータキャリアチップのように大きなデータを流す場合に正しく動作しない市販リーダー／ライター装置が多く、これまでに動作が判明しているのは協力企業の自家製装置しかなく、非常に残念な結果になっている。これは今後 PKI 利用 SIM カードなどが普及するにしたがって解決する問題とおもわれて、YRP UNL でもむしろ積極的に問題を指摘して産業界に解決を促す事項だとおもわれる。
- 個人の持ち歩く IC カードではプライバシー問題が非常に重要になっている。非接触 IC eTRON カードで、勝手に電波通信で遠隔から同定されないようにすること(privacy 保護。) が重要である。もちろん IC カードに物理スイッチをつけるという基本的な措置も必要であろうが、スイッチのきり忘れもあるだろうということで、2 重の措置としてソフトウェア上も、遠隔からの同定をできないようにする保護モードを導入した。これをもとにした試作チップでの実験を今後は行なう予定である。

以上改良点はすくないが、セキュリティ関連の問題を解決する重要な改良であった。これ以外の改良がなかったのはセキュアデータキャリアチップがすでにかなり安定した実験用チップとして実用に耐えるものになりつつあることを

示している。

（エ）クライアントライブラリの設計・実装

今年度研究・開発を行ったハイエンド型セキュアチップは，ユビキタス・コンピューティングの多様な環境での利用が想定される。多様な環境として，(1)セキュリティチップへの物理的なアクセス方法，(2)チップとの暗号通信方式ならびに認証方式，および(3)チップとの通信プロトコルの 3 つを想定し，これらを抽象化した。このうえで，これらの環境をアプリケーション側が自由に組み合わせて使えるようなクライアントライブラリの設計および実装を行った。

第 1 のセキュリティチップへの物理的なアクセス方法としては，ISO/IEC 7816 にあげられる接触型インタフェースや ISO/IEC 14443 にあげられる非接触型インタフェース，また TCP/IP などを用いたネットワーク先のチップへのアクセスが考えられる。第 2 の暗号・認証アルゴリズムとしては，対象鍵(秘密鍵)暗号・認証方式や公開鍵暗号・認証方式，これらを用いた VPN(Virtual Private Network)形成のための暗号・認証，また通信パケットに加えるパケットトレーラに使用するアルゴリズムが考えられる。第 3 のチップとの通信プロトコルとしては従来のハイエンド型セキュアチップとの互換性を保つ通信プロトコルと，新しい 16 ビット CPU を用いたセキュアデータキャリアチップとの通信用のプロトコルの両方をサポートする必要がある。クライアントライブラリは，これらの多様性環境を抽象化して構成した。

クライアントライブラリは C++を用いてオブジェクト指向で設計した。想定している多様性環境である物理的なアクセスインタフェース，暗号アルゴリズム，認証アルゴリズム，パケットトレーラの実装アルゴリズム，および通信プロトコルを，その API のみを規定した抽象クラスとして実装した。実際のアルゴリズムやプロトコルは，この抽象クラスを継承した具象クラス上で実装した（図 5-1-1)。

また，これらのクラス群の上層に，アプリケーションが選択した具象クラスを宣言する API と，チップへのアクセスコマンドを発行する API を提供するインタフェースクラスを追加した。これは，アプリケーション側が実装されている具象クラスのうち 1 つまたは複数を自由に選択して組み合わせることを可能とするためである。

本クライアントライブラリは，以下のように使用することができる。アプリケーションは，まずインタフェースクラスに対して，使用する物理的なアクセ

スインタフェース，暗号アルゴリズム，認証アルゴリズム，パケットトレーラの実装アルゴリズム，および通信プロトコルを選択し，それらの実装された具象クラスオブジェクトを生成する。次に，これらの具象クラスオブジェクトをインタフェースクラス上に実装されたメソッドを用いて宣言する。このときクライアントライブラリのインタフェースクラスは，アプリケーションから宣言された具象クラスオブジェクトを保持する。この状態でアプリケーションがインタフェースクラス上に実装されたメソッドを用いてチップへのアクセスコマンドを発行する API を呼び出すと，インタフェースクラスはすでに宣言されている通信プロトコルクラスのメソッドを呼び出してチップに送るコマンドパケットを生成し，同様に暗号ライブラリクラス・MAC ライブラリクラスのメソッドを呼び出してパケットの暗号化，パケットトレーラの追加を行い，デバイスクラスのメソッドを呼び出してチップにパケットを送信する。チップからの応答はデバイスクラスで取得し，デバイスクラスはこれをインタフェースクラスに返す。インタフェースクラスはこの後，暗号ライブラリクラス・MAC ライブラリクラスのメソッドを呼び出してパケットトレーラの確認とパケットの復号を行い，通信プロトコルクラスのメソッドを呼び出してチップからの応答データを取得する。これらの処理の後インタフェースクラスはアプリケーションに，API の返り値としてチップからの返答を返す。

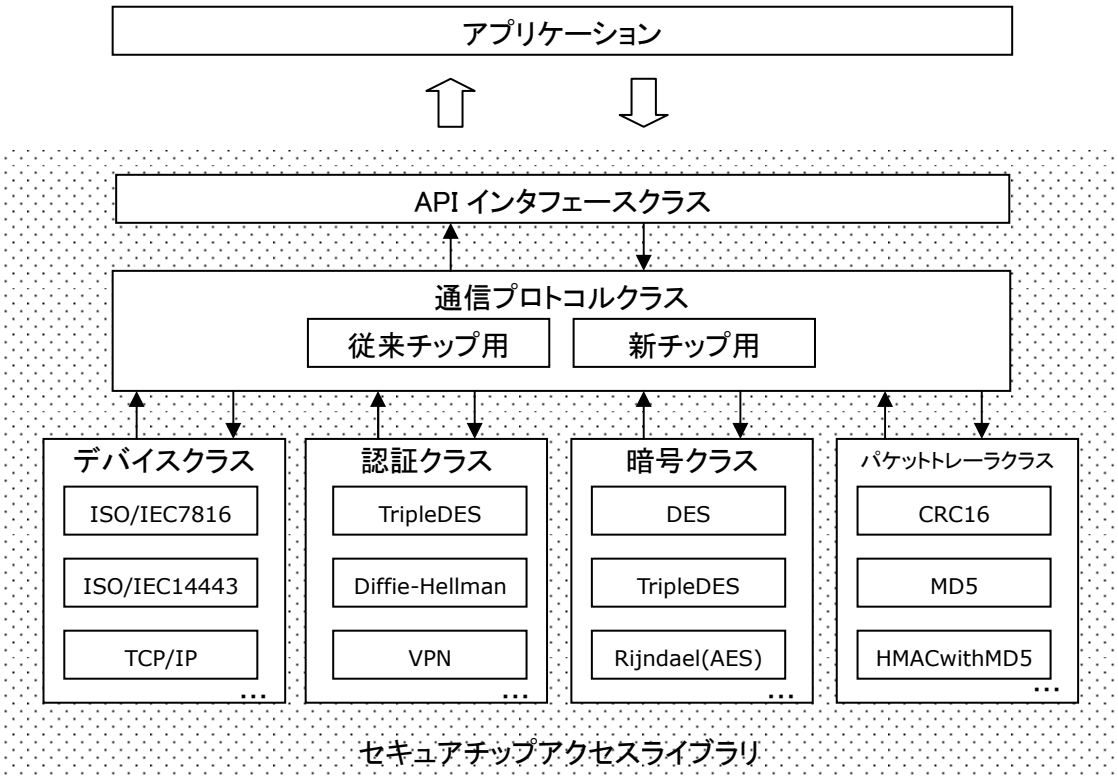


図 5-1-1　セキュアチップアクセスライブラリの構成

5-1-2　超小型セキュアチップ

（ア）研究開発の概要

本研究開発では、パッシブ型 RFID の通信状況の環境依存性に関して、水分を多く含む食品や薬剤、金属を含むパッケージ類に関して、実際に RFID をこれらのモノに貼り付け、通信試験を行った。

（イ）試験システムの概要

超小型セキュアチップの通信試験には、本事業の他サブテーマである Everything ID(ユビキタス ID)、ユビキタスコミュニケータ（UC)、アドレス解決サーバおよびユビキタス ID サーバを利用した。具体的には、それらを用いた食品のトレーサビリティ情報表示システムを構築し、超小型セキュアチップを農薬や食品に貼付し、通信試験を実施した。試験に利用したシステムは、下図の通りである。

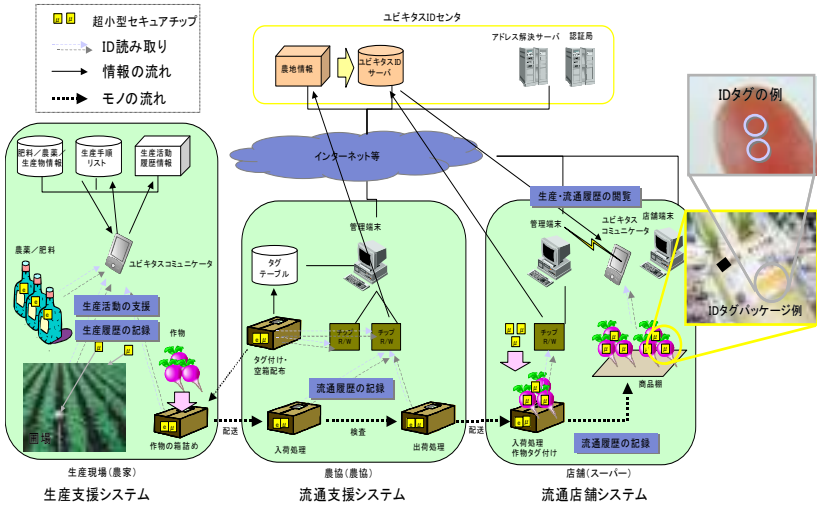


図 5-1-2 超小型セキュアチップ通信試験に用いた
「ユビキタス ID トレーサビリティシステム」

（ウ）システムの特徴

本試験システムは、前述の通り、本事業の他サブテーマにおける成果を有効

に利用したものとなっており、以下の特徴を有している。

- ID タグ（超小型セキュアチップ）
  本システムでは、チップサイズが 0.4mm 角の超小型 RFID を使用した。野菜や運搬用段ボール箱だけでなく、青果物の生産段階において、農薬、肥料等の薬剤にも RFID を貼付し、農薬のチェック等、様々な支援機能を可能とした。これらの超小型チップには、ユビキタス ID 技術を利用した ID タグ情報が格納されており、ユビキタス ID サーバを用いることで、いつでも、どこでも、商品情報の取得を可能としている。
- 生産者/消費者端末（ユビキタスコミュニケータ）
  生産者および消費者が本システムで利用する端末は、PDA サイズのユビキタスコミュニケータ（UC）である。UC は、非常にポータブルであるにもかかわらず、色々な物に貼付された超小型チップを読み取り、ユビキタス ID サーバにアクセスして商品の情報を表示することができる。
- ユビキタス ID/アドレス解決サーバ
  色々なモノに貼付された超小型チップに格納されているユビキタス ID により、アドレス解決サーバから、商品情報の格納されたユビキタス ID サーバの名前を取得することが可能である。また、ユビキタス ID サーバからは、ユビキタス ID に対応した商品情報を取得することが可能である。

（エ）試験内容

超小型セキュアチップは、色々なモノに貼付され、実フィールドで色々な環境や人により使われるものである。従って本研究では、実フィールドにおける通信精度を重視し、できる限り多様な環境における通信試験を行うこととした。

- 野菜の生産現場（農家）での試験
  実施場所：よこすか葉山農協　８農家
  実施期間：平成 15 年 9 月～H16 年 2 月
  実施内容：農薬等の薬剤に貼り付けた超小型チップの通信確認
  対象者　：農業従事者

- 食品流通現場（農協・ストア）での試験
  実施場所：よこすか葉山農協　支店集荷場
  京急ストア　平和島店、能見台店、久里浜店

実施規模：超小型チップ 約 5,000 枚
実施期間：平成 16 年 1 月 7 日～2 月 6 日
実施内容：運搬用段ボール箱に貼り付けた超小型チップの通信確認
対象者　：農協職員、ストア従業員

- 食品販売現場（ストア）での試験

実施場所：京急ストア　平和島店、能見台店、久里浜店
実施規模：超小型チップ 約 25,000 枚
実施期間：平成 16 年 1 月 8 日～2 月 6 日
実施内容：青果物等、食品に貼り付けた超小型チップの通信確認
対象者　：一般消費者

（オ）実施結果

前述の３つの環境にて実施した通信試験の結果を以下に記す。

(1) 野菜の生産現場（農家）での試験結果

野菜の生産現場ではカード型に成形した超小型チップを用いた。農薬等の薬剤に超小型チップを貼り付けたが、読み取り精度を向上させるにあたり、その方法にはいくつかの課題があった。

農薬等の薬剤への超小型チップタグ貼付においては、薬剤の形態により、貼り付け方法に工夫を加えた。農薬には袋詰め、ボトルなどの形状があり、また容器自体の材質も紙、プラスティック、アルミなど様々である（図 5-1-3）。構造的に曲げが効かないカード型のタグは、ボトル型の農薬には両面テープ等でぴったりと貼り付けることができない。また、アルミ材質のパッケージは、タグをぴったりと貼り付けると読み取りができなくなってしまった。金属のパッケージには RFID で使用する電磁誘導のエネルギーを吸収する特性があり、やはり密着して貼り付けることが困難である。これを解決するため、名札入れのような形状のケースにタグを入れ、紐で下げたり、紐に遊びをもたせ貼り付けたりすることとした（図 5-1-4）。試験対象とした農業従事者には、タグ読み取り時に金属のパッケージに近づけないで、タグを持ち上げるようにして読み込ませるように指導することで読み取り率を向上させることができた（図 5-1-5）。

図 5-1-3. 薬剤容器例

図 5-1-4 超小型チップ貼付例

図 5-1-5. 薬剤容器タグの読み取り

(2) 食品流通現場（農協・ストア）での試験結果

野菜運搬用段ボール箱への超小型チップの貼り付けは、カード型およびシール型の２種類について実施した（図 5-1-6）。

図 5-1-6. 運搬用段ボール箱への貼付例

段ボール箱に貼り付けたチップについては、カード型およびシール型のどちらについても、読み取りは比較的確実に行えた（図 5-1-7）。平面かつ水分の少ないものついては、貼り付け方法および通信精度の確保とも、比較的容易であることが実証できた。

図 5-1-7 運搬用段ボール箱タグの読み取り（左：カード型、右：シール型）

(3) 食品販売現場（ストア）での試験

食品に貼り付けた超小型チップについては、シールタイプのタグ読み取り時、貼り付け方法によっては読み取りにくいものがあった。これは、ダイコンやハーフカットにしたキャベツの芯近くにぴったりと密着して貼った場合に起きる現象で、野菜の水分により電波が吸収されることに起因している。今回はこれ

らの事象を回避するため、ダイコンの本体近くにシールを貼らず、包装ビニールの余り部分に貼る（図 5-1-8）、または、キャベツの芯近くに貼る場合でもシールを浮かして貼る（図 5-1-9）工夫を行った。

図 5-1-8. 大根への貼付例

図 5-1-9. キャベツへの貼付例

図 5-1-10. 食品タグの読み取り

食品タグの読み取りは、KIOSK 端末（図 5-1-10）やユビキタスコミュニケータ（図 5-1-10 下）をスーパーマーケット３店舗に設置し、一般消費者に実施させた。読み取り精度は、概ね良好であったが、やはり、リーダライタと超小型チップの間が水分の多い野菜で遮られると、読み取りできないケースがあった。

なお、実験期間における野菜タグの読み取り総数は５,８８５件であり、日別の読み取り回数は、図 5-1-11 の通りであった。

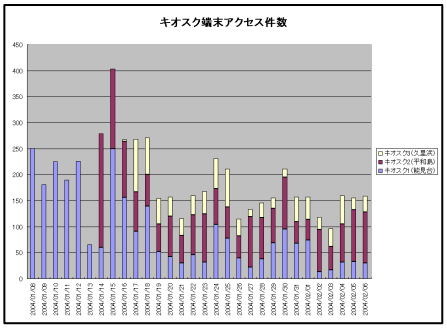


図 5-1-11. 野菜タグ読み取り件数

（カ）研究成果

超小型セキュアチップの通信試験により、RFID の貼り付け環境に応じた特性の地道な知見を得ることができた。今後様々な応用に当該技術を適用するときに十分活用できると思われる。地味な知見であったが、実用にむけては極めて重要なものであった。

まず、最も一般的に用いられると思われる超小型チップをシール形状にして貼り付ける場合についてであるが、水分の多いものへの貼り付けでは、予想通り、電波が吸収されることによる読み取り精度の低下が顕著に現れ、一般消費者が自由に使うフィールドでの利用には課題があることがわかった。対策とし

て、図 5-1-12 の通り、シール全面に糊付けするのではなく、両端部分のみ糊付けして、中央部分を浮かして貼付することにより、水分の多いモノに貼り付けた場合でも、読み取り精度を実用上、問題のないレベルまで向上させることができた。

「糊付け部分」

図１１．シール型　超小型チップの糊付け方法

また、今回の試験では、約３万枚の超小型チップをモノに貼り付けたが、その際、モノへの貼り付け作業やチップ情報の登録のための読み取り作業が、実フィールドでは非常に負担となることが明らかになった。（図 5-1-13、図 5-1-14）

図 5-1-13．超小型チップの登録作業

図 5-1-14．超小型チップの貼り付け作業

（キ）課題

今年度、超小型チップのフィールドでの通信試験を実施したことにより、超小型セキュアチップが実フィールドで使用される場合に求められるさらなる技術的課題についても確認できた。

(1)チップ登録処理、タグ貼り付けの迅速化

チップ登録作業は現在の業務にない作業であり、そのまま現場への負担増となる可能性が高い。またタグ貼り付けのため、野菜を通常は使わないビニール袋に詰め、そのビニールの上からタグを貼るため、作業時間が通常の５割増し程度となってしまう。

これは、シール形状のタグを自動登録し、ラベルプリンターのように印字も可能な仕組みを導入することにより解決できる。また、包装用ビニール自体にタグを埋め込むような技術も期待される。

(2)タグ読み取り精度の向上

今回の試験では、タグの貼り付け方法の工夫により、読み取り精度を向上させたが、今後はタグ製造時の課題として、シール内に、タグのアンテナ部分と商品側を遮断するような材質を入れるなどして、水分による電波の吸収を防ぐ工夫が必要となる。

（ク）まとめ

今年度実施した超小型セキュアチップの通信試験により、実フィールドにおいて、色々なモノに RFID を貼付する際の問題点および解決方法を明らかにすることができた。ただし、（キ）で明らかにしたような、新たな課題も見つかったため、来年度以降も引き続き、課題解決に向け、更なる研究開発を進めることとした。

5-2　基盤通信システムの研究開発

5-2-1　シームレス通信の研究

（ア）　既存ルータ混在環境におけるモバイルIP ハンドオーバの高速・高信頼化

(1)はじめに

近年、モバイル IP と呼ばれる端末の移動をサポートするプロトコルが検討さ

れている。しかし、モバイル IP ではハンドオーバに時間を要することや、それに伴うパケットの損失が発生する。そこでハンドオーバ時間を短縮し、パケットロスを抑制する方式を研究した。

(2)モバイル IP プロトコル

モバイル IP は、移動端末(MN) が本来属するホームネットワークから離れ、別なネットワークに接続した場合でもホームネットワークでのアドレス(ホームアドレス) を使い通信の継続を可能とするプロトコルである。これは、MN が移動先のネットワークで得られる IP アドレス(CoA) を、ホームネットワークに存在するホームエージェント(HA) へ登録し、HA が通信の転送を行なうことで実現される。また、通信相手(CN) に対しても CoA を通知し、HA を経由しない通信を行なうことも可能となっている(経路最適化: Route Optimization)。モバイル IP を利用することで接続先のネットワークが変っても通信を継続することが可能となるが、通常のモバイル IP では HA への登録処理(BU) が完了するまでの数秒間は通信が出来ないという問題点が存在する。この問題点を解決する方法として、高速ハンドオーバと呼ばれる方式が現在検討されている。高速ハンドオーバでは、事前に移動先における MN の CoA などの情報を移動先ネットワークのルータから取得しておき、MN のアドレス設定に要する時間短縮を図る。さらに、BU 完了までの間にハンドオーバ前のネットワークのルータ(PAR) とハンドオーバ後のルータ(NAR) の間でトンネルを設定しパケット転送を行ない、ハンドオーバ前のネットワークで取得したアドレス(PCoA) を使用して通信が行なえるようにする。このようにして高速ハンドオーバでは事前にハンドオーバ先の予測を行なうことを前提とし、MN の移動前後のネットワークに存在するルータ(PAR, NAR)の助けを借り、ハンドオーバに要する時間の短縮などを図る。しかし、通信メディアに無線 LAN(IEEE802.11) を想定すると、ハンドオーバ先を適切に予測することは難しく、高速ハンドオーバ方式の実現は困難と思われる。また、実際のネットワークでは、移動前と移動後のルータのどちらも高速ハンドオーバに対応しているとは限らない場合も考えられ、このような場合には高速ハンドオーバ機能は利用できない。

(3)提案概要

本研究では、ハンドオーバによる移動先のネットワークを予測できない状況を想定する。また、高速ハンドオーバなどに対応してない通常ルータが混在する環境を想定し、このような環境においてハンドオーバを高速に行なう方式を研究する。さらに、無線 LAN を用いた場合にはハンドオーバ時に最低でも数百

ミリ秒は通信が不可となるように、下位レイヤにおいて発生する避けられない通信不能な時間に対処するために、本方式でもハンドオーバ時のパケットバッファリングについても検討する。

前述の前提条件をふまえ、本方式では通常ルータの配下に MN が移動した際には、MN が自らトンネルの設定を行なう。また、移動先のルータに応じ、PAR あるいは HA でのパケットバッファリング動作を切り替える。

高速ハンドオーバと同様に MN は常にレイヤ 2 のリンク状態を監視し移動の検出を行なう。移動先ネットワークを事前に予測できないため、MN はリンクダウン状態からリンクアップを検出し、Router Solicitation(RtSol)メッセージを送信して CoA 取得などを試みる。また、PAR, NAR の各ルータが本方式に対応したルータ(PARp, NARp)か通常のルータ(PARc, NARc)かにより図 5-2-1 のように動作を行なう。

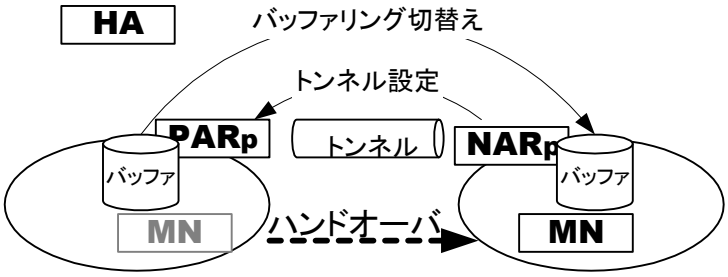


図5-2-1　考案方式の概要

(a) 対応ルータ間の移動: 高速ハンドオーバと同様に PARp と NARp 間でトンネルを設定するが、この際 NARp から PARp へトンネル設定を行なう。

(b) 対応ルータから通常ルータへの移動: MN 自身が CoA 取得後に PARp へトンネルを設定する。その後、PARp のバッファからパケットを受け取り、加えて BU 完了まで PCoA を使用した通信を再開する。また、HA への BU メッセージで HA にバッファリングの開始を指示し、MN のハンドオーバ完了後は HA がバッファリングを行なう。

(c) 通常ルータから対応ルータへの移動: NARp から HA へトンネルを設定し、同様にして NARp は HA のバッファから受取り MN へ送り、加えて BU 完了までの間 PCoA を使用した通信を再開する。そして、HA でのバッファリングから NARp でのバッファリングに切り替える。

(d) 通常ルータ間の移動: 通常のモバイル IP のハンドオーバと同様になる。但し、HA への BU 完了後に HA からバッファリングパケットを受け取る。

(4)　今後の検討課題

(4－1)セキュリティ

モバイル IP や高速ハンドオーバでは、本来のネットワーク上の位置とは異なる場所へ通信を転送する。このため第三者によるなりすましなどを避けるためのセキュリティに関する検討がされている。本方式においても同様に、なりすましなどの攻撃から保護するための対策を行なう必要がある。

(4－2)バッファリング

パケットロスに対しパケットのバッファリングを行なうが、TCP のように輻輳や再送制御を行なっている場合には、これらの制御に悪影響を及ぼすことも考えられる。また VoIP 等のようにリアルタイム性が重要とされる場合には、バッファリングにより大きな遅延が生じることも好ましくない。そのため、バッファリングの利用方法や対象となるパケットの量や時間などを検討する必要がある。

5-2-2　ユビキタス情報提供・制御用プロトコルの研究

以下の通り、ネットワークアーキテクチャとプロトコルのレイヤ構造に関して基礎研究を行い、それを元にユビキタス情報提供・制御用プロトコルのデータリンク層／トランスポート層／セッション層（セキュリティ層）の仕様策定から実装までを行った。

(ア)ネットワークアーキテクチャ

ユビキタスネットワークに求められる機能要件として、

- 厳密なリアルタイム性が確保できること
- 通信方式は、コネクションレスであること
- 制御系ネットワークなので、データ長は短くすること

がある。

以上より、基本的なネットワークアーキテクチャは、アクセス方式はトークンパッシングとし、トポロジはバスおよびスターを基本とすることとした。

次にパケット長であるが、ECHONET, LonWorks, ARCNET, EC-NET 等の制御系ネットワークプロトコルでは、データ長は 256byte 未満のものが幾どである。これは、制御系ネットワークの特質としてメッセージ長が短いという点から来ているのであろう。さらに、256 = 8bit ということでインターフェイスチップの回路コストを抑えるのにも多少なりとも貢献している。パケット構成を考

えると、データ部の長さを 256byte にする方式とヘッダ込みのパケット全体の長さを 256byte にする方式がある。今回はデータの最大長よりも処理速度/メモリ利用効率を重視するため、パケット長は全体で、max 255byte(ヘッダ 8byte + データ 247byte)とする

本ネットワークでは、収容するノードの数が膨大になるため、機能追加やバグフィックスをネットワーク経由で行なう必要がある。プログラムは当然 256byte にはおさまらないので、TCP のような上位レイヤでのパケットの分割/組立を行なうことになる。また、ネットワークの効率的な利用の為のコマンドのマクロ化等を考慮して、上位レイヤでのデータの分割/組立を行うようにする。

次にアドレスについて。今回のネットワークでは、個々のノードに 128bit のユニークなアドレス(uID)が組み込まれている。しかし、これをそのまま使うのはルーティング負荷やノード管理の負荷の点で得策ではない。そこで、上記物理アドレスとは別に、IP アドレス相当の論理 ID を個々のノードに付与して管理負荷を減らすことにする。既存の制御系ネットワークにおいては、アドレスフィールドは 8bit のものが多い。ノード数的には、最大 254 ノード程度となる。本ネットワークの応用例の一つである家庭内ネットワークを想定した場合、ノード数は概算で 600 ノード程度となり 8bit では不足する。かといって、600 ノードを単一のネットワークに収容するのは非効率であり、またリアルタイム性にも影響を与えることになる。そこで、ネットワークは複数のドメインに分割できるものとするすなわち、アドレス空間は階層構造を持てるようにする。ここで、例えば空調機器などは温度センサ/湿度センサと空調機器本体のように複合機能を持っているため、物理的には 1 台のノードであっても論理的には複数のノードを収容していることになる。これを解決するには、以下の案があげられる。

案 1：複数個の物理インターフェイスを持つ

→簡単だが、あまり現実的ではない

案 2：物理インターフェイスは 1 つにする

→有るべき姿だと思うが、どうやって実現するのか？

そこで、複数のドメインに分割でき、かつ上記課題を解決するため、案 2 を採用する。案 2 を採用した場合、さらに 2 つの案が挙げられる。

案 2-1：論理ノード識別用にアドレスフィールドを確保する

→安直。冗長。拡張性に乏しい

案 2-2：論理ノード識別はアドレスフィールドとは別のフィールドで定義

→プロトコルのレイヤ構造の観点では、こちらが妥当。拡張性あり。

よって、アドレスについては、メイン識別用とドメイン内での物理ノード識別用という空間に分割し、それぞれ 8bit の空間を割り当てる(最大で約 64k の論理ノードを収容可能)ことにする。ドメインの分割は自由であるが、部屋単位にドメインを分割しておくと管理しやすいし、リアルタイム性も確保しやすいであろう。

ドメインの分割には、ルータを使用するここで言う「ルータ」とは IP 網で広く使われているルータとは違い、Layer2 switch 相当の機能である。

センサ類の情報は、自ドメイン内および制御系サーバへのマルチキャストとする

具体的には、図 5-2-2 の構成を想定する。

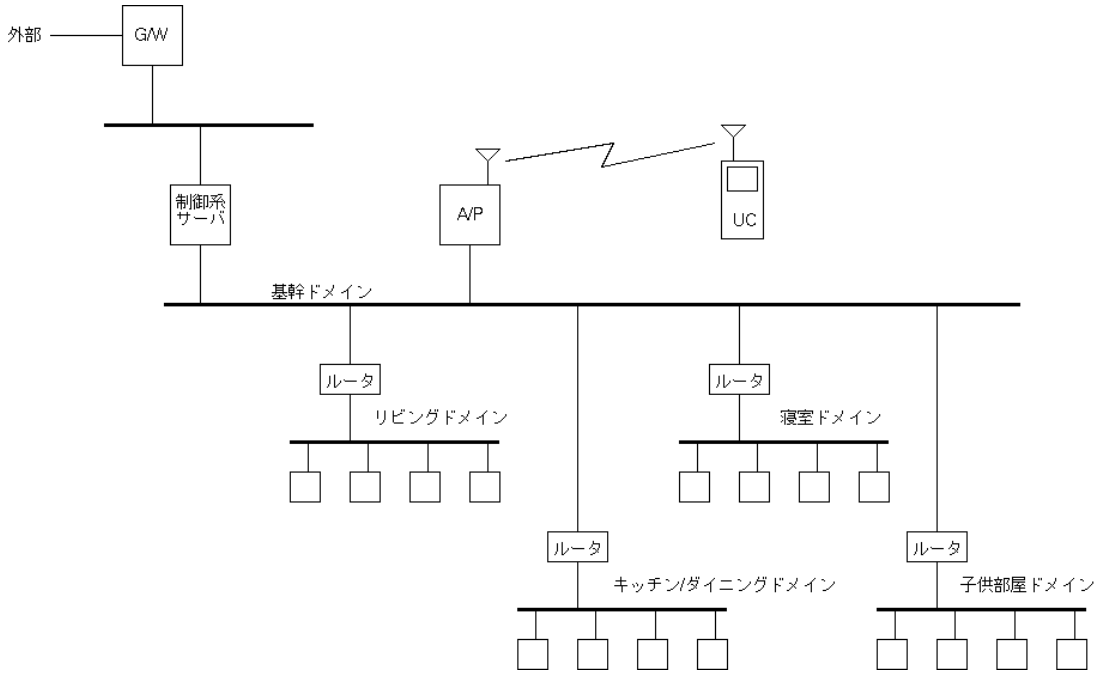


図 5-2-2 ネットワークアーキテクチャ

(イ) プロトコルのレイヤ構造

OSI 参照モデルでは通信プロトコルを 7 階層のレイヤに分割している。そこで、本プロトコルの特質を元に、どのようなレイヤ構造が望ましいのかを以下の通り検討した。

OSI 参照モデルのうち、アプリケーション層(第 7 層)/データリンク層(第 2

層)/物理層(第 1 層)は必須の階層なので、プレゼンテーション層(第 6 層)からネットワーク層(第 3 層)までの要否を検討した。

1. プレゼンテーション層(第 6 層)
   このレイヤでは、データのコード変換、暗号化、データ圧縮／伸長、ファイル転送等を行う。このうち、本プロトコルで必要な機能は暗号化あたりであるが、暗号化処理はもっと下位のレイヤおこなうつもりなので、それを考慮すると、本レイヤは不要となる。
2. セッション層(第 5 層)
   このレイヤでは、セッションコネクションの設定／解放や送信権の制御、データの区切りの制御などを行う。本プロトコルはコネクションレス型のネットワークであり、セッションを確立して制御を行うなどといったことは想定していないため、本レイヤは不要となる。
3. トランスポート層(第 4 層)
   このレイヤでは、高信頼で低コストな通信手段を提供することが目的となる。具体的には、多重化／逆多重化、分割／組立、連結／分離などである。さらに、end-end での誤り検出などである。本プロトコルではパケットの多重化／逆多重化、データの多重化／逆多重化を行なうので、この処理を行なうレイヤが必要となる。
   そこで、トランスポート層は、データの多重化／逆多重化を行なうサブレイヤとパケットの多重化／逆多重化を行なうサブレイヤの 2 つのサブレイヤで構成することとする。
4. ネットワーク層(第 3 層)
   このレイヤでは、データ伝送時の経路制御やフロー制御などを行う。本プロトコルのアドレス体系は階層構造を持つものを想定しているため、機能的には本レイヤのものが含まれることになるが、なるべく軽いプロトコルにしたいので、経路制御機能はデータリンク層へ追いやることにする。また、本プロトコルはアクセス方式としてトークンパッシングを採用しているので、プロトコルレベルでのフロー制御は不要である。よって、本レイヤは不要とする。
5. セキュリティ層(OSI 該当無し)
   ユビキタスネットワークは従来のネットワークと比較してより普段の人間

生活に密着したネットワークであるため、プライベートな情報を取り扱う機会が多くなる。本レイヤが無いとプライバシーが侵害されたり生命/財産が危機に面する可能性があるため、セキュリティ機能が必須となる。
本プロトコルではリアルタイム性を追求しており、ネットワーク管理の基礎となる基本プロトコル群はデータリンク層で処理することにしているため、比較的処理負荷の重いセキュリティ層はデータリンク層の上に定義する。

以上より、OSI 参照モデルと比較した場合の本プロトコルのレイヤ構造は、図 5-2-3 となる。

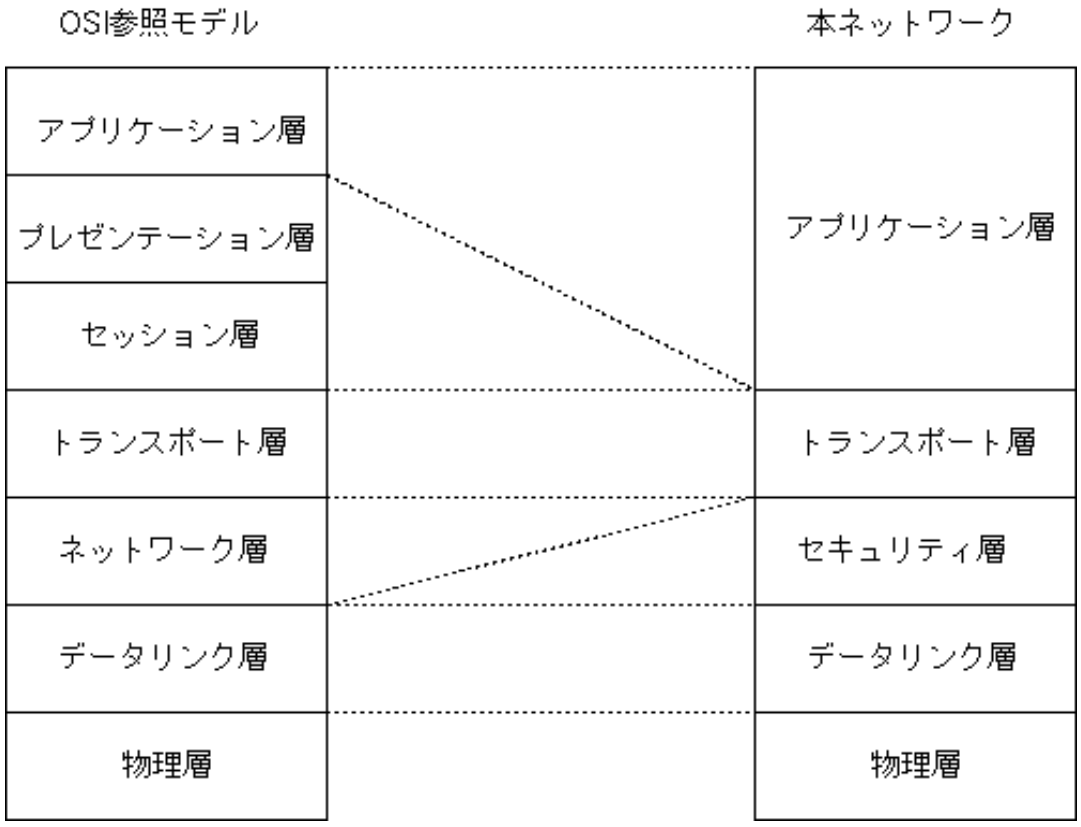


図 5-2-3 プロトコルのレイヤ構造

また、基本的なパケットフォーマットの詳細を図 5-2-4 に示す。各フィールドの上の数値はバイト数である。

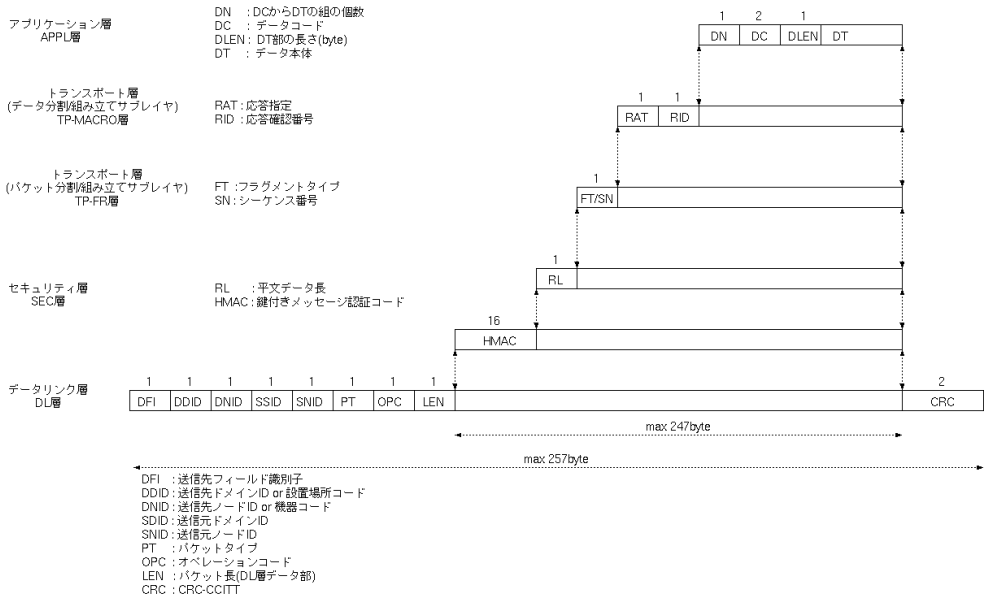


図 5-2-4 パケットフォーマット

(ウ) データリンク層

データリンク層のプロトコルには既存のトークンパッシング方式を改良したプロトコルを開発して実装した。通信形態は、ユニキャストとマルチキャスト（ブロードキャストを含む）をサポートした。また、パケットヘッダの特定のフィールドを検査することによって、優先制御を行うリアルタイム通信と、ベストエフォート通信の処理ができるようにした。ネットワーク上の論理アドレスについては、機器同士が協調してアドレスを管理できるアルゴリズムを開発した。機器が削除された場合でも、論理的に近くにいる機器がタイムアウトすることによって削除されたことを検知するアルゴリズムを開発した。また、本レイヤを LSI 化することでシステム全体の処理速度の向上を図った。

データリンク層におけるパケットフォーマットを図 5-2-5 のようにする。また、データリンク層パケットパラメータを表 5-2-1 に示す。

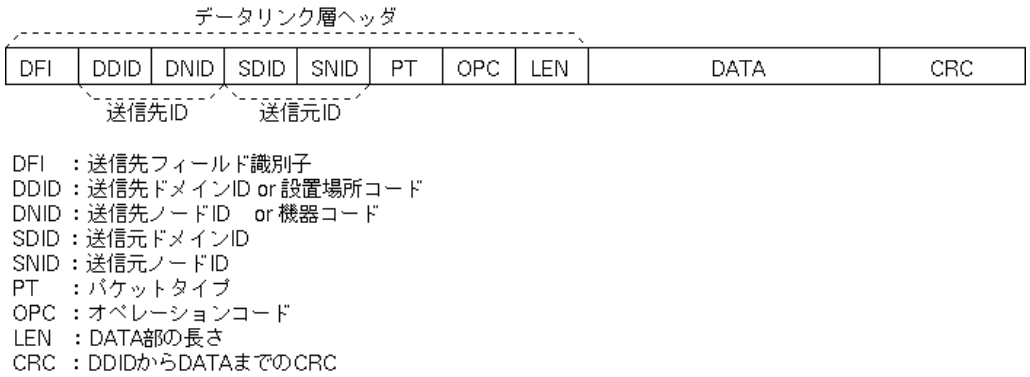


図 5-2-5 データリンク層パケットフォーマット

表 5-2-1 データリンク層パケットパラメータ

名称	機能	サイズ(byte)	備考
DFI	送信先フィールド識別子	1	
DDID	表2を参照	1	
DNID	表2を参照	1	
SDID	送信元ドメインID	1	
SNID	送信元ノードID	1	
PT	パケットタイプ	1	
OPC	オペレーションコード	1	
LEN	パケット長	1	
DATA	データ	0～247	DATA部のサイズ
CRC	CRC	2	DFIからDATAまでに対するCRC-CCITT

以下、各パラメータについて説明する。

1. DFI(送信先フィールド識別子)

   従来仕様は、ネットワークの自動構築などの網管理に重点をおいた仕様であった。しかし、本来ユーザはノードの論理的な ID がどうであるかを気にすることなく、「この部屋の空調の温度を下げたい」とか「この部屋の照明を暗くしたい」というような要求をするのが自然である。

   基本的には、ドメインと部屋は一対一であることを想定しているが、常にそのような設置が行なわれるとは限らない。従来の仕様では、このような想定を逸脱した設置が行なわれた場合に、意図したネットワーキングができない可能性がある。

   そこで、ロケーションコードと機器コードを仕様として定義しておき、これらの値をヘッダに使用して送信先情報を「どこにある」+「どんな機器」という型式でも指定できるようにする。

   補足

   - 通常モードのパケットは、全ドメインへのブロードキャストと等価

であるため、ルータは、送信先フィールド識別子が 0x00 の場合は上り下りとも無条件で中継するのが(ルータの負荷的に)良さそうに思える。しかし、この場合配下ドメインがブロードキャストストームとなる可能性がある。そこで、ルータに配下ドメインのノードのロケーションコードと機器コードのキャッシュテーブル(ロケーションキャッシュテーブル)を用意しておき、通常モードのパケット受信時には送信先 (DDID/DFID)がテーブルにヒットしたパケットだけをキューイングして配下ドメインに流すようにすることでトラフィック量を低減させた。

- 同一「ロケーション」中に複数の同一「機器」がある場合の識別は、ペイロード中に uID を載せて特定することで実現する。逆に、ペイロード中に uID が格納されていない場合は、その部屋の当該機器全てが処理対象となる。

2. DDID/DNID

前項でも説明したように、送信先フィールドである DDID/DNID は、DFI の値によって意味合いが違ってくる。その関係を、表 5-2-2 に示す。

表 5-2-2 DFI 値による送信先 ID の位置付け

DFI値	モード名	DDID値	DNID値	備考
0x00	通常モード	ロケーションコード	機器コード	通常運用において、「あの部屋の照明を(窓を、空調を,etc.)操作したい」といった処理を、プロトコルレベルで表現するために使用する。(ソフトウェアdefault値)
0x01	網管理モード	送信先ドメインID	送信先ノードID	データリンク層におけるネットワーク自動構築時に使用する。(ハードウェアdefault値)

ここで、ロケーションコードとは、その機器が設置されている場所を 8bit でコード化したものである。例えば、洗面所は 0x05、玄関は 0x02 などのように定義する。

機器コードとは、自ノードがサポートする機能等を 8bit でコード化したものである。例えば、空調機は 0x10、人感センサーは 0x96 などのように定義する。

3. SDID/SNID

ドメイン ID/ノード ID ともに、未定義時の値は 0x00 とする。ブロードキャストパケットとは、送信先ノード ID(DNID)が 0xFF のパケットとする。このため、ノード ID 値は、暗黙のうちに 1～254 となる。

ドメイン ID 処理の負荷を無くすため、ドメイン ID は、当該ルータの(ローカル側の)ノード ID 値と同一の値とする。

特定ドメインへのブロードキャスト(自ドメインを含む)は、ドメイン ID をその値にして、ノード ID 値を 0xFF にする。

送信先ドメイン ID が 0x00(未定義)の場合は、当該ドメイン内のノードとしか通信できないこととする。

4. PT(パケットタイプ)

リアルタイム性を追求するため、複数プロトコルに対応できるフォーマットを規定する。パケット種別の識別子として、PT(Packet Type)フィールド 8bit を定義する。その際、パケット受信時にデータリンク層レベルで処理するかどうかを判定するため、 3 つのカテゴリに分類する。現時点で定義済の値は、表 5-2-3 の通りである。

表 5-2-3 PT

データリンク層で処理する	
0x00	トークン
0x01	プローブ
0x02	勧誘
0x03	勧誘結果通知
0x04	ドメインID広告
0x05	ドメインマスター
0x06	next hop更新指示
上位レイヤにパケットを渡す	
0x10	通常パケット
0x11	自己紹介
0x12	減設通知
0x??	仮想ノード間通信 (?? : 90, A0, A1, A2)
OPCの値(※)によっては、データリンク層で処理する	
0x20	ARP (ノードID→uID)
0x21	RAPRP (uID→ノードID)
0x22	ping

※OPC=0x10(要求)の時は、データリンク層で処理。OPC=0x20(応答：肯定)の時は上位レイヤにパケットを渡す。

上記のうち、受信時の処理をデータリンク層レベルで行なう PT=0x00～0x06 のパケットについては、図４で定義したパケットフォーマットのうち、データリンク層ヘッダ以降は独自のフォーマット(直接データを格納する)になる。

5. OPC(オペレーションコード)

パケットのオペレーションコード用の識別子 OPC(Operation Code)フィールド 8bit をパケットヘッダに定義する。8bit のうち、上位 4bit で通知/要求/応答を定義、下位 4bit でそれぞれのオペレーション内容を定義する。

表 5-2-4 OPC

値	機能
0x00	通知 （一般）
0x04	通知 （機能アラーム）
0x08	通知 （ソフトウェア異常）
0x09	通知 （対象機器異常）
0x10	要求 （read, get）
0x11	要求 （write, set）
0x20	応答 （肯定）
0x21	応答 （否定:未定義領域）
0x22	応答 （否定:権限無し）
0x23	応答 （否定:宛先無し）
0x24	応答 （否定:受信バッファフル）
0x25	応答 （否定:ソフトウェアアップデート）
0x26	応答 （否定:アップデートビジー）
0x27	応答 （否定:その他）

6. LEN(パケット長)

パケット長は、DATA 部の長さを byte 数で表す。範囲は、0～247byte とする。(CRC 部は本レイヤで終端する。)

7. DATA(データ)

上位層のデータを格納する。サイズは、0～255byte である。

8. CRC(CRC)

通常パケットにおいては、CRC-CCITT にてパケットの正常性を検査する。 生成多項式 = $X^{16} + X^{12} + X^5 + 1$ 。

範囲は、DFI から DATA 部まで。

エラーを検出した場合は、応答を返さずエラーカウンタを加算する（パケッ

ト全体が信用できないため、誰に応答を返したら良いのか判断できない)。
CRC はデータリンク層で終端するため、上位レイヤには渡さない。

(エ) トランスポート層

本レイヤは、パケットの分割／組み立てを行なう FR サブレイヤと、データの多重化／逆多重化処理を行なう MACRO サブレイヤに分かれる。

1. TP-FR サブレイヤ

このサブレイヤでは、プログラム転送など 257byte 以上の大量のデータ転送を実現するために、パケットの分割／組み立てを行なう。
（FR は、Fragment and Reassembly の頭文字をとったもの。）

送信時には、上位レイヤ(TP-MACRO 層)から受信したデータを 257byte(厳密には 222byte)以下になるように分割して下位レイヤ(セキュリティ層)に渡す。
受信時には、下位レイヤ(SEC 層)から受信したデータを必要に応じて結合して上位レイヤ(TP-MACRO 層)に渡す。

制御系ネットワークにおいては、パケットの分割／組み立てが必要な大量のデータ転送が発生するのはプログラム転送など限られたケースしかないため、本サブレイヤのヘッダは 1byte に最適化してある。図 5-2-6 に、そのイメージを示す。

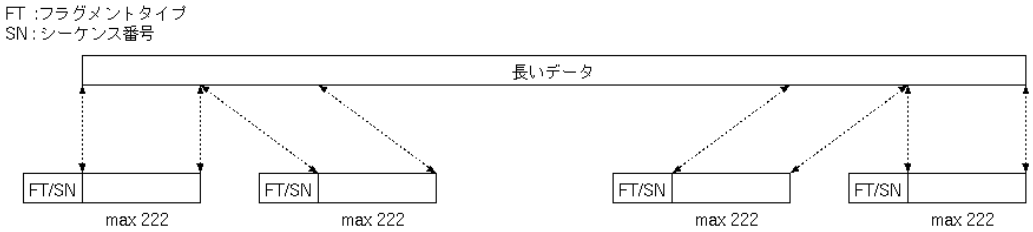


図 5-2-6　TP-FR 層

表 5-2-1 TP-FR サブレイヤパケットパラメータ

名称	機能	サイズ(bit)	備考
FT	フラグメントタイ	上位3bit	
SN	シーケンス番号	下位5bit	modulo32

以下、各パラメータについて説明する。

・ 1 FT

Fragment Type。3bit。

当該パケットの、フラグメント状況を示す。bit 割り当ては、表 5-2-6 を参照。

表 5-2-2 FT の割り当てと機能

FT値			機能
bit7	bit6	bit5	
0	1	0	BOM (先頭パケット。Beginning of Message)
0	0	0	COM (中間パケット。Continuation of Message)
0	0	1	EOM (最終パケット。End of Message)
0	1	1	SSM (単一パケット。Single Segment Message)
1	*	*	ABORT (組み立て中止通知)

送信ノードがなんらかの理由でフラグメントされたパケットの送信を途中で中止したい場合、FT 値を ABORT のコードにしたパケットを送信する。受信ノードでは、FT 値が ABORT のコードのパケットを受信したら、組み立て中の処理を中止してリストを初期化する。

・ SN

Sequence Number。5bit。

本サブレイヤで分割されたパケット毎にシーケンス番号を付与する(モジュロ 32)。先頭パケット(BOM)あるいは単一パケット(SSM)の場合は、all'0'とすること。

受信ノードでは、この SN 値を元にパケットから元のデータを組み立てていく。パケット受信に失敗したときは、この SN 番号を NACK の引き数としてサーバ等の配布元に通知して再送要求を行なう。

・ 分割／組み立てのイメージ

FT/SN を使ったパケットの分割／組み立てのイメージは、図 5-2-7

のようになる。当該パケットの、フラグメント状況を示す。 bit 割り当ては、表 5-2-6 を参照。

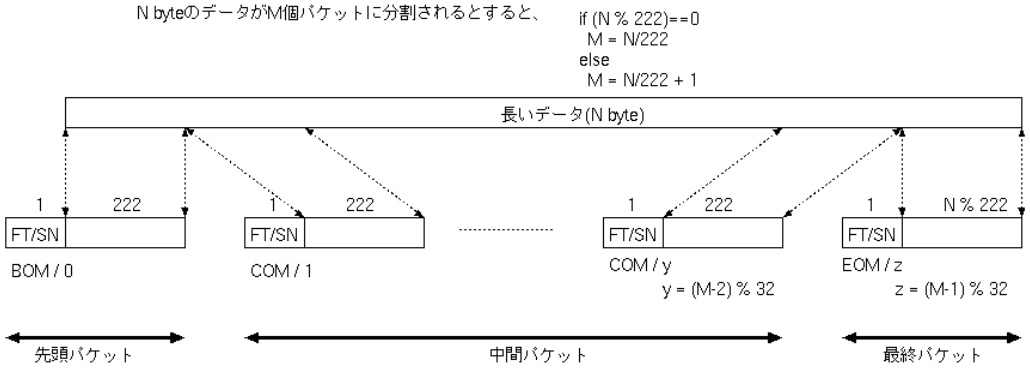


図 5-2-1 分割／組み立てのイメージ

2. TP-MACRO サブレイヤ

主にコマンドのマクロ化等に使用することを想定しており、アプリケーション層で行なう自律分散協調動作を実現するためのコマンドの学習用に使用する。

このレイヤではパケットサイズの制限は無くなり、巨大なデータを扱うことが可能である。

図 5-2-8 に、そのイメージを示す。

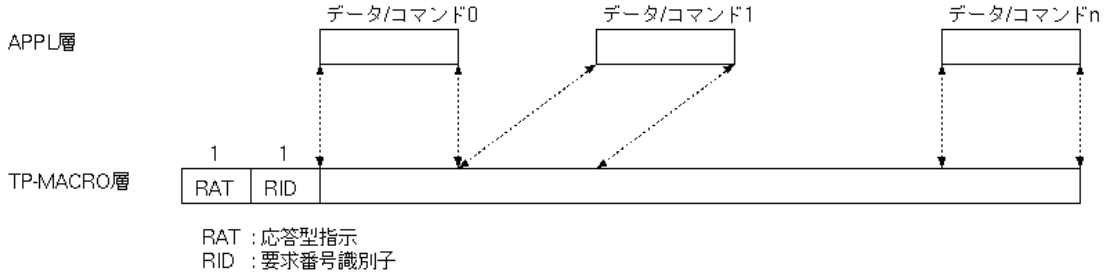


図 5-2-2 TP-MACRO 層

表 5-2-3 TP-MACRO サブレイヤパケットパラメータ

名称	機能	サイズ (byte)	備考
RAT	応答型識別子	1	required acknowledge type
RID	要求番号識別子	1	request ID

以下、各パラメータについて説明する。

- RAT

Required Acknowledge Type。1byte。

送信ノードが要求している、acknowledge のタイプおよび返送タイミングをコード化したもの。上位 4bit で要求する ack の種類を、下位 4bit でタイミングを指示する。その詳細を表 5-2-8 に示す。

表 5-2-4 RAT コード詳細

値	機能
0x0*	No ACK/NAK is required
0x10	Required ACK at my-turn. (Don't send me NAK)
0x11	Required ACK at your-turn. (Don't send me NAK)
0x20	Required NAK at my-turn. (Don't send me ACK)
0x21	Required NAK at your-turn. (Don't send me ACK)
0x30	Required ACK or NAK at my-turn. (Send me anything)
0x31	Required ACK or NAK at your-turn. (Send me anything)

my-turn:この場合、直ちに ACK/NACK を返送する必要がある。通常、このパターンが多い。

your-turn:この場合、受信ノードにトークンがまわって来たときに、ACK/NAK を返送する。ブロードキャスト/マルチキャストで write/read する時など、ACK/NAK が複数ノードから同時に上がって来るとパケット衝突がおきてしまうので、この返送タイミングを使って順次 ACK/NAK が返って来るようにする。

例えば、プログラムの一括転送をする場合、

- サーバがプログラムをブロードキャストする
  →同時に ACK/NAK を返してはいけない
- いちいち ACK は返さないで良い
- でも再送するためには NAK はきちんと返して欲しい

等の条件があるので、0x21(Required NAK at your-turn)が最適な RAT となる。通常は、0x10/0x20 が多い。

- RID

Request ID。1byte。

送信ノードにおいて、要求に対する応答を管理する ID。送信ノードが Read 要求と Write 要求を連続して送信した場合の応答を判断する場合などに使用する。

(オ) セッション層

セキュリティや認証のための機能を提供した。相互認証と同時に鍵交換を行い、セッション中は暗号通信が行なわれるセキュアセッション機能、リアルタイム応答が要求される場合のライトウェイトセッション機能を備えた。また、階層構造をもつネットワーク構成においてセキュリティの強度を上げるために、各ドメイン単位で異なる鍵を使用するなどの配慮も行った。なお、目標設定時はセッション層でセキュリティや認証機能を実現する予定であったが、セキュリティの強度を上げるため、これらの処理はネットワーク層の上層に実装した。

1. 前提条件

本システムは、複数のドメインから構成される２階層のアーキテクチャとなっている。本システムの主な応用例の一つとして、住宅への適用がある。この場合、基本的には１台のルータが１つの部屋を管理することを想定しているが、１台のルータが複数の部屋を管理したり、逆に複数のルータで１つの部屋を管理する場合もある。ネットワーキング自体はドメイン ID とノード ID を基本とした論理アドレスで行なうが、実空間とのマッピングを行なうために、プロトコル上で前述の「ロケーションコード」を使用している。ルータはこれら「ロケーション」のマスターとなるが、セキュリティの基本単位(鍵を共有する単位)も「ロケーション」とする。

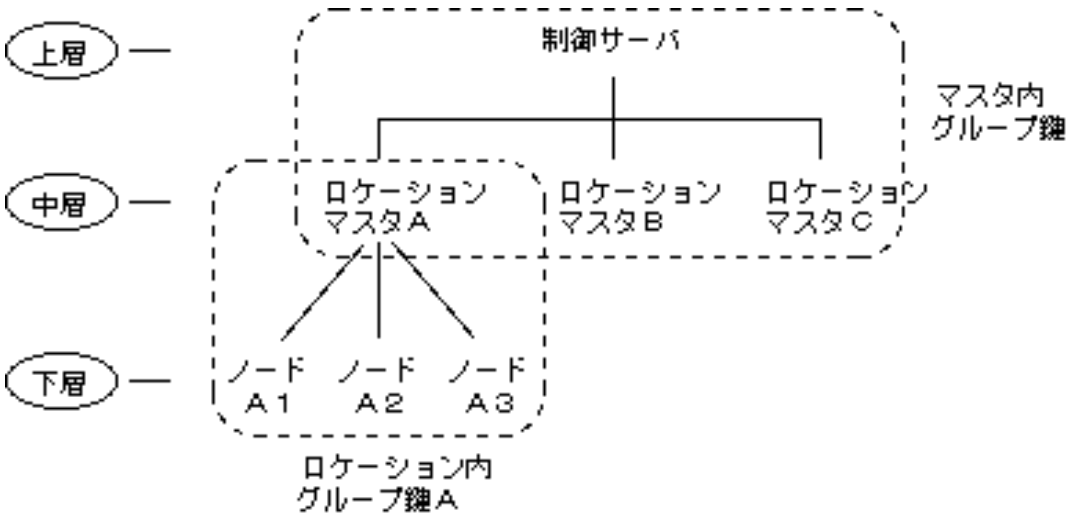


図 5-2-10 セキュリティ階層とグループ鍵共有

本システム内では，下記 3 つのセキュリティ階層で構成されている．

① 制御サーバ（上層）：
上中層間で暗号化通信を行うための暗号鍵（マスタ内グループ鍵）を生成し，中層の全ロケーションマスタに配送する上位ディストリビュータ．

② 各ロケーションマスタ（中層）：
中下層で暗号化通信を行うための暗号鍵（ロケーション内グループ鍵）を生成し，下層の当該ロケーションノードに配送する下位ディストリビュータ．当該ノードから外部ロケーション／制御サーバへのセキュリティパケット，他上中層から当該空間のノードへのセキュリティパケットをマスタ内グループ鍵／ロケーション内グループ鍵で再構築する．

③ 各ロケーションノード（下層）：
グループ鍵の生成／配付機能はない．当該空間のマスタから受け取った共有暗号鍵にて，空間内及び外部空間へのセキュリティパケットを送受信する．

2. セキュリティパケット

送信側のセキュリティパケット構築手順と，受信側のセキュリティパケット検証／復号手順を示す．送信側と受信側は，共通のグループ鍵を共有している．異なるロケーションノード間，または上層と下層間といった，互いに共有のグループ鍵を持たない場合の通信においては，中層が仲介を行う。

本節で用いる各種暗号関数を次のように表記する．

$E_K[X]$　共通鍵 K を用いた，データ X の共通鍵暗号化値

$D_K[X]$　共通鍵 K を用いた，暗号化データ X の共通鍵復号値

1 セキュリティパケット構築手順（送信側）

セキュリティパケット構築は大きく以下の手順に従う．

- 使い捨ての暗号化鍵を計算し，同鍵により DATA 部を暗号化
- 使い捨ての MAC 鍵を計算し，同鍵により暗号化 DATA に対する MAC 計算
- データリンク層ヘッダ，暗号化 DATA，MAC をビット連結（セキュリティパケット完成）

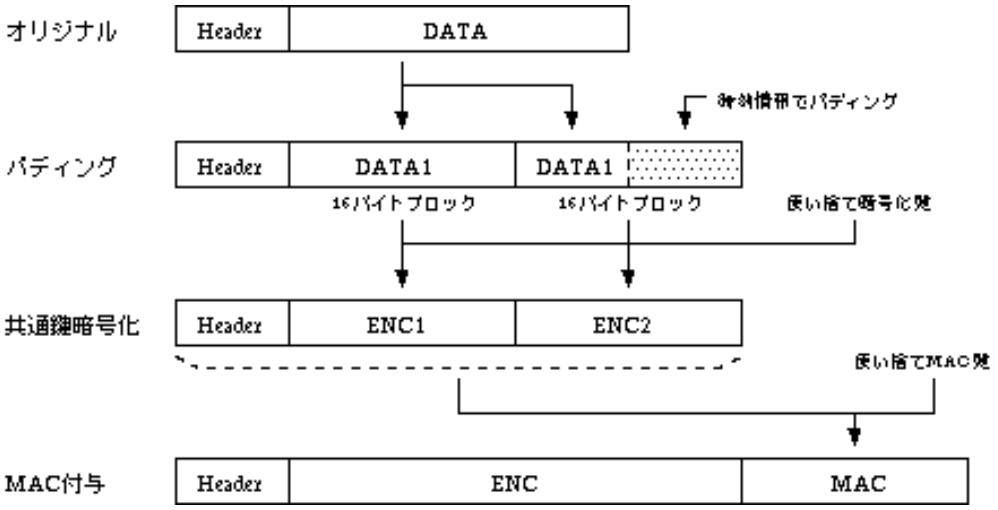
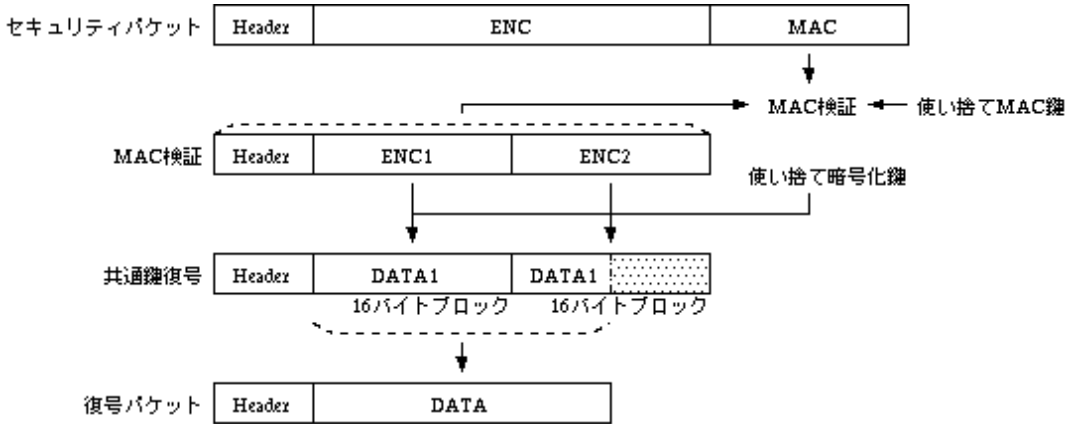


図 5-2-11 セキュリティパケット構築手順



を抽

図 5-2-12 セキュリティパケット復号／検証手順

3 セキュリティパケット伝達の流れ

相手が解釈可能なセキュリティパケットを届けるには，送受信者双方で同一の時刻情報とグループ鍵を共有する必要がある．それ故，表 9 に示すような，上記要件を満たさない状況では，中層の当該ロケーションマスタが仲介して，受信ノードが解釈できるパケットに再構築する．

表 5-2-9 セキュリティパケットの伝達の再構築

通信種別	パケット再構築の必要箇所
上層・下層ノード間	下層ノードの所属ロケーションマスタ（1箇所）
異種ロケーションの下層ノード間	送信側ロケーションマスタ，受信側ロケーションマスタ（2箇所）

ここでは，Worst Case と見積もられる，異なるロケーションの下層ノード間でのセキュリティパケット伝達の流れを説明する．今，ロケーション A の下層ノードがロケーション B の下層ノードにセキュリティパケットを送る場合を考える．

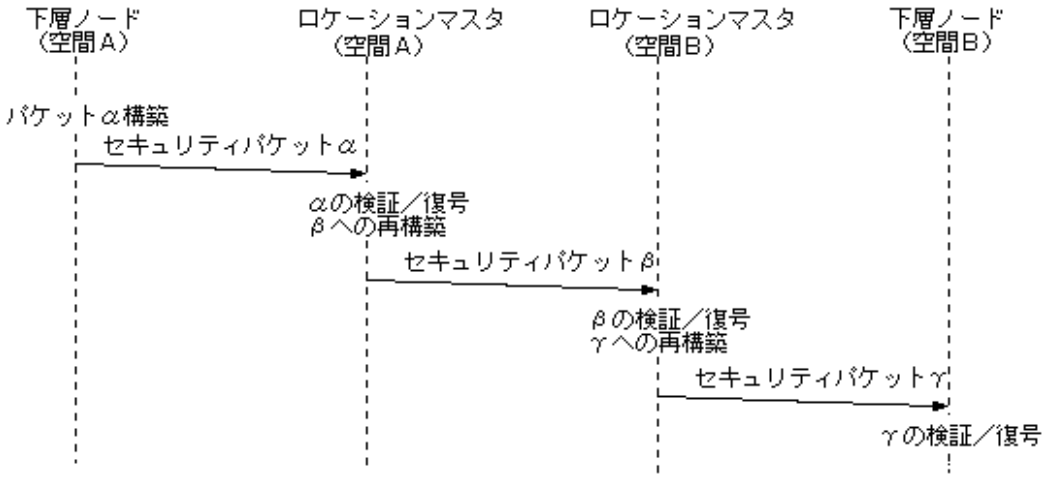


図 5-2-13 異種ロケーションノード間のセキュリティパケット伝達

1. ロケーションAの下層ノードは，空間Aでの時刻情報と空間Aのロケーション内グループ鍵を用いてセキュリティパケットαを構築し，送信する．
2. 空間Aのロケーションマスタは，データリンク層ヘッダを見て，異空間へのパケットであることを確認すると，以下の手順でパケットの再構築を行う．
   ① 空間Aでの時刻情報と空間Aのロケーション内グループ鍵を用いて，セキュリティパケットαの検証および復号を行う．
   ② 上中層で共有の時刻情報およびマスタ内グループ鍵を用いて，セキュリティパケットαと同様のデータリンク層ヘッダと DATAに対するセキュリティパケットβを構築，送信する．
3. 空間Bのロケーションマスタは，データリンク層ヘッダを見て，自空間ノードへのパケットであることを確認すると，以下の手順でパケットの再構築を行う．
   ① 上中層で共有の時刻情報およびマスタ内グループ鍵を用いて，セキュリティパケットβの検証および復号を行う．
   ② 空間Bでの時刻情報と空間Bのロケーション内グループ鍵を用いて，

セキュリティパケットβと同様のデータリンク層ヘッダと DATAに対するセキュリティパケットγを構築，送信する．

4. ロケーションBの下層ノードは，空間Bでの時刻情報と空間Bのロケーション内グループ鍵を用いて，セキュリティパケットγの検証および復号を行う．

（カ） まとめ

以上の通り、ユビキタス情報提供・制御用プロトコルの研究に関して、平成１５年度はネットワークアーキテクチャとプロトコルのレイヤ構造に関して基礎研究を行い、それを元にユビキタス情報提供・制御用プロトコルのデータリンク層／トランスポート層／セッション層（セキュリティ層）の仕様策定から実装までを行った。その際、データリンク層はハードなリアルタイム性が要求されるため、ハードウェア（LSI）化した。それ以外のレイヤは、ソフトウェア上で順次実装を行った。また、サーバやルータの実装や、本プロトコルの性能評価／機能評価等については、平成１６年度に行う予定である。

5-2-3 実世界研究・Everything ID 研究

（ア）研究開発の目的

ユビキタスコンピューティング環境では、実世界環境に埋め込まれたセンサやアクティブタグなどの多様な超小型チップが、多様なネットワークを介して大量に相互接続される。それらの超小型チップから得られる膨大な量の実世界情報を有効活用するためには、超小型チップに対応する実世界情報を瞬時に検索し、取得する仕組みが必要となる。このような大量の超小型チップに関連する実世界情報の管理･検索を実現するために、以下の条件を満たす方式を開発する必要がある。

- 情報量に対するスケーラビリティを備えた方式
- 十分な検索スピードを実現可能な方式

そこで、本研究開発では、膨大な数の超小型チップのネットワーク接続にも対応可能な実世界情報管理検索方式の検討を行う。具体的には、超小型チップにヒモ付く実世界情報を管理･検索するためのディレクトリ構造及び検索メカニズムについて、そのアーキテクチャと管理検索方式の設計を行う。

（イ）今年度の取り組み範囲

今年度の取り組みは、以下のとおりである。

- 基本アーキテクチャの設計
- 番号分散管理方式の設計
- 実世界情報検索方式の設計

まずは実世界情報管理検索の基本的な思想の確立、基本的な機能の設計を行い、様々な応用システムの要求に対応するための拡張性に富んだシステムアーキテクチャの基盤作りを目指す。

（ウ） 方式設計

（１）基本アーキテクチャ

ユビキタスコンピューティング環境では、膨大な数の超小型チップが多様ネットワークを通じて相互接続され、それらのチップから得られた実世界情報を蓄積し、いつでも自由に利用できることが望まれる。このとき、超小型チップにヒモ付けられる情報を分散管理する手法として、以下の 2 つの方式が考えられる。

A） 超小型チップ内に情報を格納する方式

B） ネットワークで接続されたサーバに情報を格納する方式

A)方式は、現在普及しつつある IC カードで利用されている方式であり、ネットワークに接続することなく情報を取得できるという利点がある。しかし、超小型チップは IC カード等と比較してリソースに乏しく、情報の格納領域やアクセス制御機能を持てない場合が多い。従って、超小型チップの情報管理アーキテクチャとしては、B)方式のようなサーバ蓄積型の方式が適切であると考えられる。これを実現するためには、それぞれの超小型チップに一意な番号を付与し、番号をキーにしてサーバから情報を取得するための仕組みが必要になる。そこで、本検討では、図 5-2-14 に示すアーキテクチャに基づいた番号管理検索方式の設計を行った。

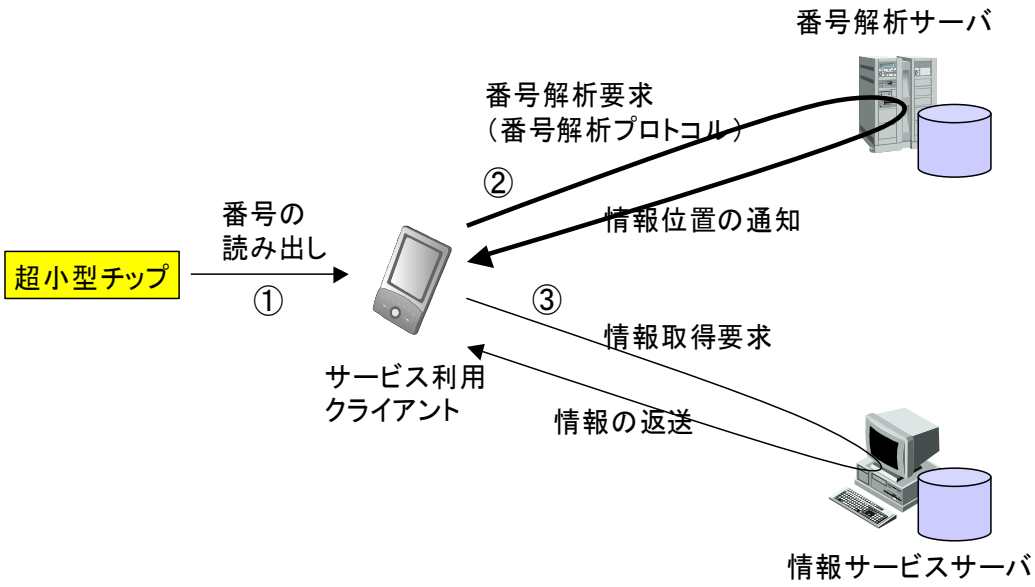


図 5-2-14　基本アーキテクチャ

本アーキテクチャでは、２つのステップにより番号から実世界情報の取得を行う。第一ステップでは、番号をメタ情報（情報の格納位置）に変換する。第二ステップでは、メタ情報を元に実世界情報の実体の取得を行う。以下に、本アーキテクチャの基本構成要素を以下に示す。

- 超小型チップ
  - モノに添付される。
  - あらかじめ一意な番号が付与されている。
- サービス利用クライアント

超小型チップの実世界情報を取得するためのデバイスであり、ＰＣや携帯電話などを想定する。

  - 超小型チップから番号を読み出し、番号解析サーバに送信することで取得したい実世界情報の位置を把握し、情報の取得を行う。

- 番号解析サーバ
  - 番号からそれにヒモ付く実世界情報の格納位置を解決するディレクトリサービスを提供する。
- 情報サービスサーバ
  - 番号にヒモ付く実世界情報の実体が格納されている。

２ステップ処理とすることで、管理形態や目的に応じて情報の実体を格納するサーバを分散配置することが可能となる。また、分散配置された情報サーバはそれぞれ異なる事業者や個人が管理することができるため、管理コストの観点からもスケーラビリティの高い構成と言える。

次に、超小型チップからそれにヒモ付く実世界情報を取得するまでのフローを示す。

① 番号の読み出し
- 無線インタフェースなどを利用し、超小型チップから番号を読み出す。

② 番号解析要求
- 読み出した番号を番号解析サーバに送信する。
- 番号解析サーバから番号にヒモ付いた情報の格納位置を受信する。

③ 情報取得要求
- 番号と②で取得した情報の格納位置に基づき、情報サービスサーバから実世界情報のダウンロードを行う。

（２）分散管理方式

基本アーキテクチャでは、超小型チップのリソースに依存することなく超小型チップにヒモ付く実世界情報を管理することが可能である。しかし、膨大な数の超小型チップがネットワークに接続される環境下では、単独の番号解析サーバが全ての超小型チップの番号を管理することは非現実的であり、番号の分散管理方式について検討を行う必要がある。100 億レベルの番号解析を実現するためには、データベースサーバを単にクラスタリングするような物理的な分散化ではなく、番号の属性に応じた論理的な分散構造を設計する必要がある。また、本研究で提案している番号体系はメタコード体系であり、JAN コードや ISBN を始めとする既存の番号体系を包含できる設計となっている。従って、このような既存の番号体系にも適用可能な分散管理方式とする必要がある。図 5-2-15 に、番号の分散管理方式の基本的な考え方を示す。

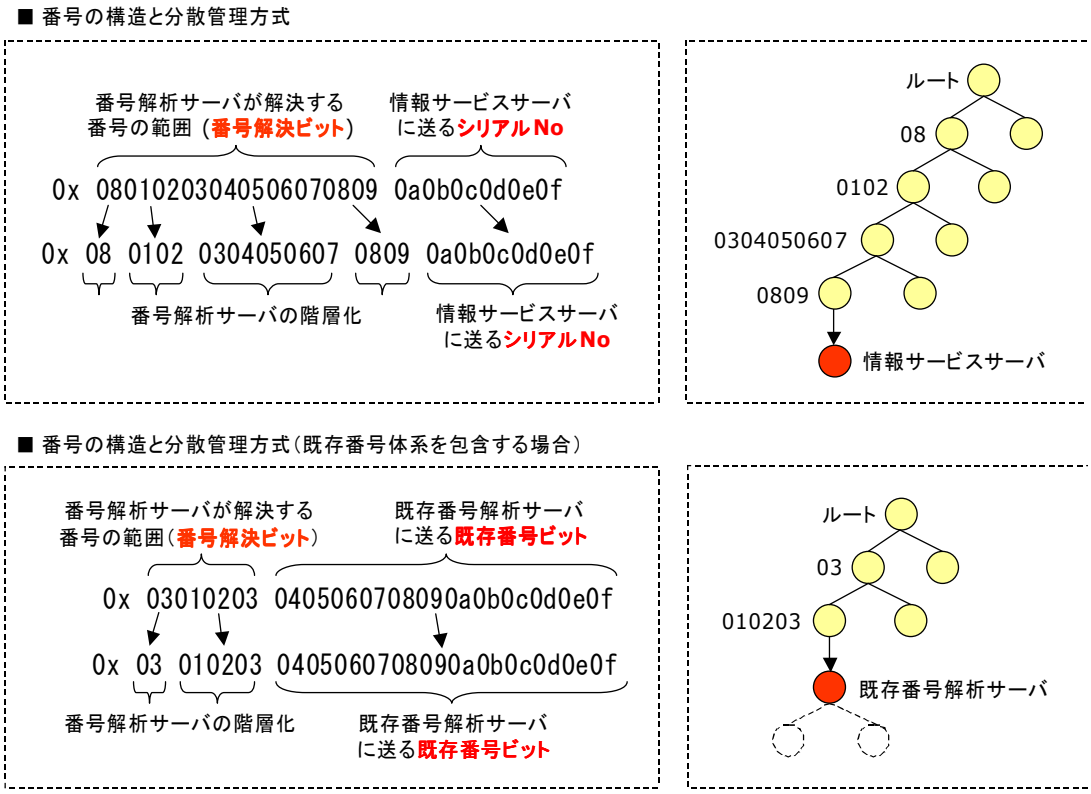


図 5-2-15　番号の分散管理方式

番号は、番号解析サーバが解決する番号の範囲である“番号解決ビット”と、情報サービスサーバに送信する“シリアル No”から構成される。一方、番号の管理ディレクトリは木構造となっており、上位に位置する番号解析サーバから順番に番号解決ビットの解決を行っていく。自分が持つデータベースで番号解決ビットの全ビットを解決できない場合には、残りのビットの解決は下位の番号解析サーバに委譲される。このとき、それぞれの番号解析サーバは別の事業者や個人が管理することが可能である。サービス利用クライアントは、番号解決ビットを最後まで解決することで情報サービスサーバの位置を取得し、得られた位置情報とシリアル No に基づき、番号にヒモ付いた情報を取得する。

また、既存番号体系を包含する番号の場合には、シリアル No にあたる部分に既存番号体系で規定される既存番号ビットが設定されている。この場合は、番号解決ビットの解決により既存番号解析サーバの位置が得られるものとし、その後の番号解決は既存番号解析サーバのアーキテクチャにて行われる構成とする。

次に、識別解決サーバにおける分散管理方式の具体的方式について述べる。

本研究では、論理的な分散を実現し、かつ既存番号にも適用可能な 2 つの方式の検討を行った。

1. クラスによる分散

番号からヒモ付け情報を取得するための代表的な分散ディレクトリサービスに、インターネットで広く利用されている DNS（Domain Name Service）がある。DNS の逆引き検索では、IP アドレスを 8bit 単位で解決することで DNS 名への変換を行う。IPv4 アドレス空間が約 42 億であることを考えると、ある一定のスケーラビリティを実現可能な方式と言える。そこで、番号をある一定の単位で解決することで、その単位毎に分散ディレクトリを形成する方式を提案する。方式概略を図 5-2-16 に示す。

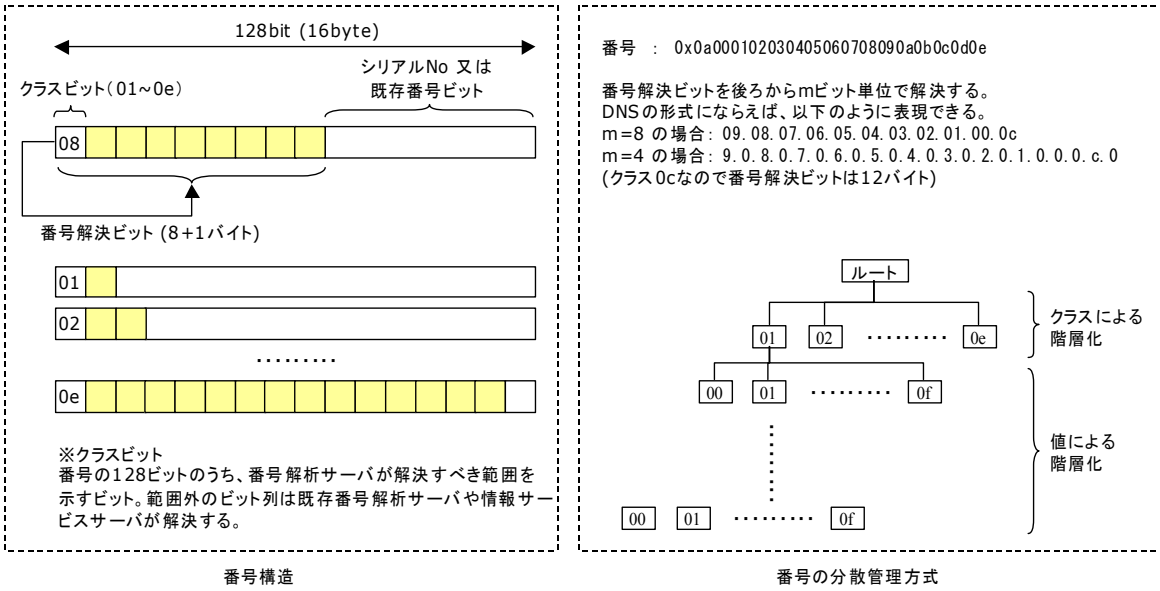


図 5-2-16　クラスによる分散

- 番号構造
  - ✓ 先頭 n ビット（図では n=8）をクラスビットとして定義し、クラスビットにより番号解析サーバが解決すべきビット範囲が固定される構造とする。
  - ✓ 既存番号体系を利用する場合には、番号解決ビットの次のビットから最終ビットが既存番号ビットとなる。既存番号ビットは本アーキテクチャ内では解決されず、既存番号解析サーバに解決が委

譲される。この場合、番号解析サーバはヒモ付けられた情報の格納位置ではなく、既存番号解析サーバの位置を返すことになる。

- 番号の分散管理方式
  - ✓ 番号解決ビットを m ビット単位（図では m=8）で固定的に階層化し、各階層に番号解析サーバを分散設置する。

2. ビットマスクによる分散

番号解決ビット長や階層化を固定的ではなく、必要に応じて動的に変更できるメカニズムとして、ビットマスクを利用する方式である。ビットマスクにより動的に階層構造を構築することが可能となり、より多様なアプリケーションの要求に対応可能な方式であると言える。方式概略を図 5-2-17 に示す。

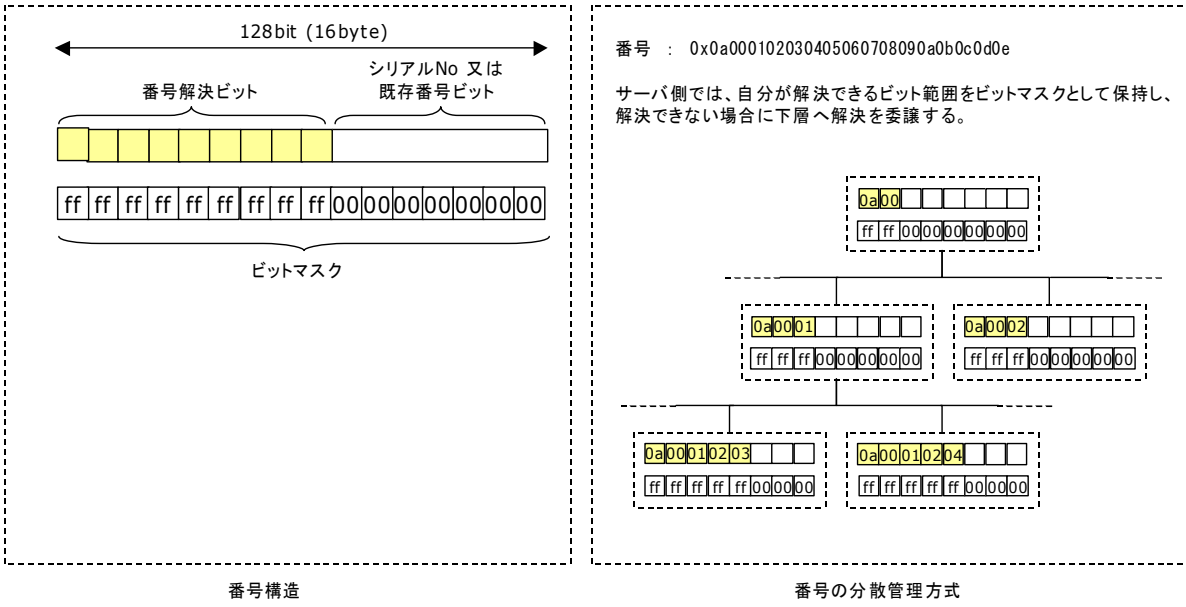


図 5-2-17　ビットマスクによる分散

- **番号**構造
  - ✓ 番号解析サーバへの解決要求送信時に、サービス利用クライアントがビットマスクにより番号解決ビットを指定する。(図では上位 72 ビットを指定)
- 番号の分散管理方式
  - ✓ 番号解析サーバでは、自分が解決可能なビット範囲をビットマスクとして保持しており、解決できない場合には下層の番号解析サ

ーバに解決が委譲される。

- ✓ 各番号解析サーバがビットマスクを自由に設定できる。また、番号の値ごとにビットマスクを指定できるようにすることで動的かつフレキシブルな階層構造を実現でき、より柔軟に番号解析サーバを分散配置できる。

これら２つの方式の特徴をまとめると、表 5-2-10 のようになる。

表 5-2-10　各方式の特性

評価項目	A)方式 クラスによる分散	B)方式 ビットマスクによる分散
フレキシビリティ	×	○
実装時の重さ	○	△
既存技術の使用	△	×

- **フレキシ**ビリティ
  - A)方式は、番号解決ビットや階層構造が固定的であり、適切な分散構造をとれない恐れがある。
  - B)方式は、番号解決ビットや階層構造を動的にフレキシブルに設定可能であり、各階層ごとに適切な解決ディレクトリを構成することが可能である。
- 実装時の重さ
  - A)方式は、固定的な処理なので処理が軽い。
  - B)方式は、ビット演算による動的な解決パスの生成が必要となるため、比較的処理が重い。また、要求パラメータにマスクビットが含まれるために、ネットワークプロトコルの観点でも A)方式と比較して非効率的である。
- 既存技術の使用
  - A)方式は、DNS などの既存ディレクトリソフトウェアを利用した実装が可能である。
  - B)方式は、ソフトウェアを最初から構築する必要がある。

本研究では、以下の理由により、B)方式を採用することとした。

- **ビットマスクにより各番号**解析サーバが管理する番号範囲を個別に指定できれば、各階層で固定的なビット範囲を規定することなく、フレキシブルな分散構造を構築できる。フレキシブルな分散管理により、よりきめ細かな空間割り当てが実現可能となり、番号空間の有効利用の観点で望ましい。例えば、IPv4 の DNS の逆引きは、8bit 単位でアドレス解決／分散管理するモデルであったために、アドレス有効利用のために途中から CIDR (Classless Inter-Domain Routing)や VLSM (Variable Length Subnet Mask)による複雑な分散管理方式を導入せざろうえなくなり、実装が非常に複雑になったという経緯がある。
- 方式設計の観点で言うと、拡張性は選択した方式により決定されてしまう一方で、実装時の重さについてはハードウェアやソフトウェアの改良による対応が可能である。
- 既存技術の使用による開発の効率化は、新技術にとって重要なテーマではあるが、本研究におけるスケーラビリティ、リアルタイム性といった課題と比較した場合に優先すべき項目とは言いがたい。

（３）情報検索方式

超小型チップからヒモ付く実世界情報を取得するとき、ユーザは分散管理された情報の位置を意識することなく、いつでもどこからでも同様の手続きにより情報を検索･参照できる必要がある。

(3-1) 方式概要

本アーキテクチャでは２ステップにより実世界情報の検索を行う。各ステップにおける情報検索方式の概要を以下に示す。

- 第一ステップ（番号解析要求）

  番号解析サーバは分散管理されているため、分散された番号解析サーバから目的の番号を管理している番号解析サーバを検索しなくてはならない。検索の基本的なフローは図 5-2-18 のとおりになる。

  番号解析サーバが３段階に階層化されており、目的の番号を番号解析サーバ C が管理している場合のフローである。この場合、サービス利用クライアントはまず識別解決サーバ A（ルートサーバ）に番号解析要求を

送信する。このとき、サービス利用クライアントは何らかの手段によりルートサーバのアドレスを知っているものとする。ルートサーバでは番号の一部のビットから、次に検索すべき番号解析サーバとしてサーバ B のアドレスを返送する。サービス利用クライアントがサーバ B へ番号解析要求を送信すると、サーバ B は番号の一部のビットから、次に検索すべき番号解析サーバとしてサーバ C のアドレスを返送する。最後にサーバ C へ番号解析要求を送信すると、サーバ C は番号の一部のビットから、番号にヒモ付けられた情報を格納している情報サービスサーバのアドレスを返送する。

以上のように、番号解析サーバの役割は、自らが持つ番号データベースを検索することで、番号に関するアドレス情報を要求元に返信することである。アドレス情報としては、例えば以下のものを利用することが可能である。

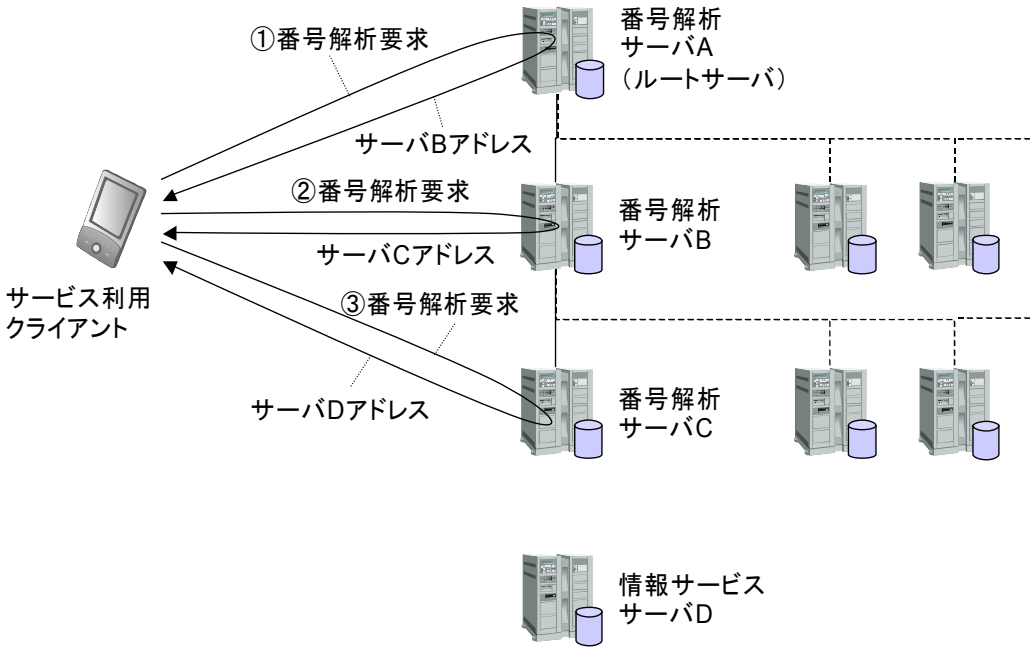


図 5-2-18　解析フロー概要

- 128/256/384/512bit 番号
- IPv4 アドレス(32bit)

- IPv6 アドレス(128bit)
- URL
- e-mail アドレス
- 電話番号

このように、インターネット以外の通信手段による情報の取得もサポートすることで、より多様なネットワークへの対応を実現する。

- 第二ステップ（情報取得要求）

第一ステップで取得した情報サービスサーバのアドレスに基づき、番号にヒモ付いた情報のダウンロードを行う。情報サービスサーバに送信する情報は、典型的には番号そのものであるが、番号以外のアプリケーション固有の ID などを利用してもよい。

なお、情報の記述方式については、多様な言語や実世界情報の記述が可能で、かつサービス利用クライアントが共通的に解釈できる標準的な仕様を設計する必要がある。これについては今後の課題である。

(3-2) 応用検索

- カスケード検索

本アーキテクチャは２ステップによる情報検索方式であるが、その応用としてカスケード型の検索方式も検討した。カスケード型検索には次の２つのパターンがある。

A) 番号解析のカスケード

番号解析要求に対して、番号解析サーバは自分が管理する番号データベースを検索し、アドレス情報を返信する。番号解析サーバが多段に

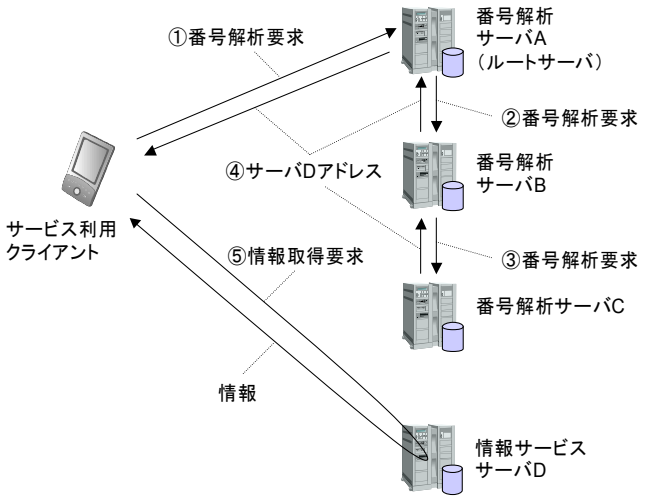


図 5-2-19 番号解析のカスケード

階層化されている場合には、番号解析要求／応答のやりとりを階層数分だけ行う必要があるため、オーバヘッドが大きくなる。そこで、番号解析サーバ自身が下層の番号解析サーバとの通信を行うカスケード検索の仕組みを取り入れた。カスケード検索のフローを図 5-2-19 に示す。

① サービス利用クライアントが番号解析要求を送信する。
② 番号解析サーバ A（ルートサーバ）は、番号に関連付いたアドレス情報を検索する。検索されたアドレス情報が番号解析サーバアドレス（サーバ B アドレス）であるため、自らがクライアントとなって番号解析要求をサーバ B に送信する。
③ サーバ B でも同様に検索されたアドレス情報が番号解析サーバアドレスであるため、サーバ C に番号解析要求を送信する。
④ サーバ C で検索されたアドレス情報が情報サービスサーバアドレスであるため、サーバ C はカスケード検索を行わず、アドレス情報をサーバ B、サーバ A を経由してサービス利用クライアントに返送する。
⑤ サービス利用クライアントはサーバ D から情報取得を行う。

番号解析のカスケードには、以下のようなメリットがある。

- サービス利用クライアントのネットワークが PHS 等の狭い帯域を利用している場合には、繰り返しサーバへ要求を送信するオーバヘッドを無視できない場合がある。カスケード検索を行うことにより、番号解析要求が 1 回で済むことから、高速化が期待できる。
- サーバ間の通信では、あらかじめ VPN などのセキュアセションを張ることが可能であり、認証時のオーバヘッドを抑えることが可能となる。

B) 情報サービスサーバへのカスケード

番号解析サーバだけでなく、情報サービスサーバまでカスケード接続を行う。この場合、番号解析要求の返答として、番号にヒモ付けられた情報の実体が返送される。フローを図 5-2-20 に示す。

情報サービスサーバまでカスケードを行う場合、クライアントサーバ

間のトランザクションをさらに減らすことが可能になるが、この場合には番号解析サーバと情報サービスサーバ間の通信プロトコルをあら

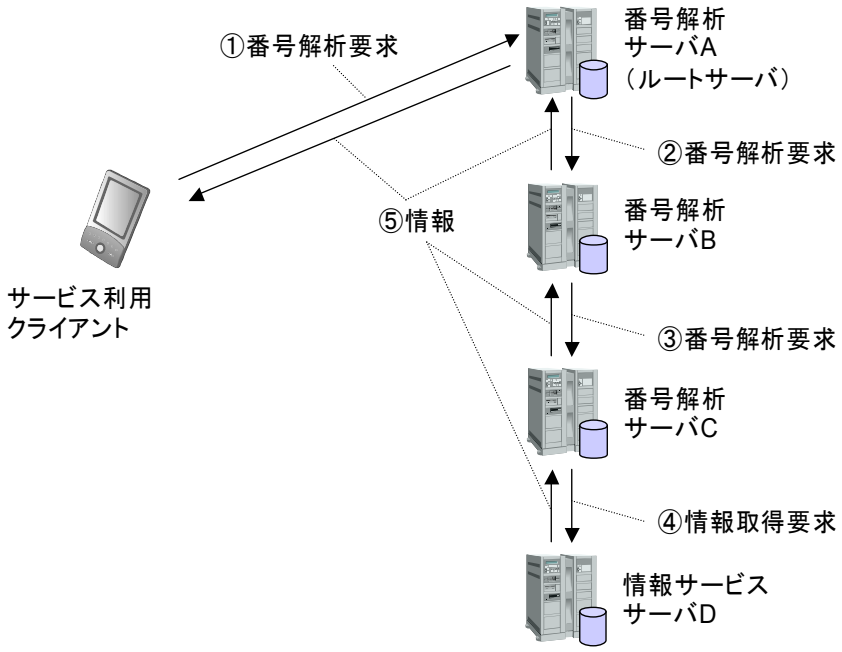


図 5-2-20 情報サービスサーバへのカスケード

かじめ取り決めておく必要がある。

- エリアサーバ検索

  サービス利用クライアントに代わり、番号解析を行うエリアサーバを設置する構成である。エリアサーバが設置されているネットワークでは、エリアサーバに番号解析要求を送信すれば、最終的な検索結果のみが返送される。従って、クライアントのリソースやネットワーク環境が貧弱である場合にも、より高速な検索が可能となる。

  また、本アーキテクチャにおいては、通常はサービス利用クライアントに番号解析ルートサーバのアドレスが登録されていることが前提となるが、自分の近くに存在するエリアサーバを自らが発見する機能を搭載できれば、番号解析サーバのアドレスを知らないクライアントであっても番号の解決が可能になるというメリットがある。

  さらに、セキュリティ的な観点から言うと、エリアサーバはプロキシの役割を果たすため、サービス利用クライアントのアドレスを秘匿することが可能であり、プライバシ保護が重要となりうるケースで有効である

と言える。

図 5-2-21 に、エリアサーバを設置した場合のフローを示す。

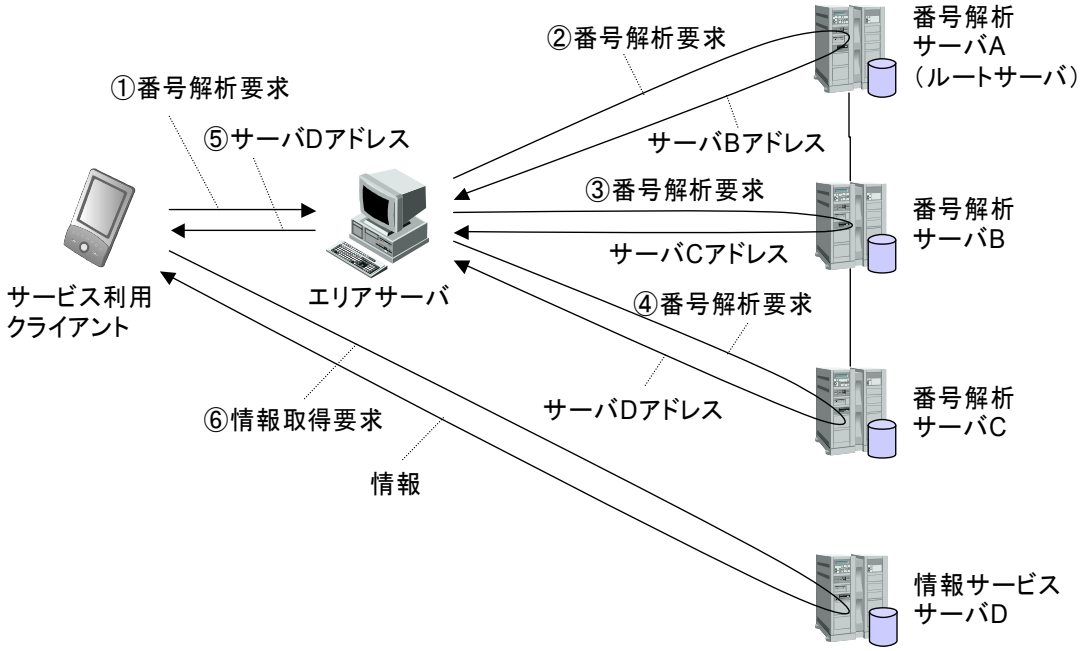


図 5-2-21　エリアサーバ

（エ） 結論

（１）研究開発の成果

本研究開発では、超小型チップに番号を割り振り、番号からそれに関連付いた実世界情報を検索するための基本アーキテクチャの設計を行った。また、番号の管理方式として、ビットマスクを利用した分散管理方式を採用し、より拡張性の高いスケーラブルな分散ディレクトリを設計できた。情報の検索方式としては、カスケード方式などの応用的な検索方式を設計し、より多様な要求に対応するための基盤を確立できた。

本研究開発により、以下の項目の実現が可能となった。

- 一意な番号が付与された超小型チップから、超小型チップに関連付いた実世界情報の格納位置を検索し、情報を取得できる。
- 番号解析サーバを論理的に分散配置し、膨大な数の番号を同一アーキテクチャ内で管理できる。
- クライアント側のリソースやネットワーク環境に応じて、適切な検索方式を選択できる。

（２）課題

今年度の研究により、機能的観点から最低限の実世界情報検索機能を実現できたが、運用性、利便性、安全性の観点では多くの課題が残されている。次年度以降は、性能･効率やセキュリティ等のより実用化に向けた要求に対応していくための研究開発を行う。以下に、残存している課題を示す。

- 検索スピードの向上
  - キャッシュ方式
  - プロトコル設計の改良
- 解析サーバの管理方式
  - 情報の登録･削除･更新の方式
  - 効率的な管理手法の設計
- 実世界情報の表現方式
  - 実世界情報の表現方式の標準化

5-3　ユーザノードシステムの研究開発

5-3-1 ユーザノードシステムの開発

ユビキタスコンピューティング環境において利用者（ユーザ）が直接接する事になるユーザノードを、研究全体の標準プラットホームのアーキテクチャをベースにしたもので実現した。このノードは移動ノードと固定ノードという２種類に大別でき、それぞれにおいて開発を完了した。移動ノードに関してはユビキタスコミュニケータ（以下 UC）と UC-Phone の２種類を開発した。

（ア）ユビキタスコミュニケータの定義

最初にその名称の由来となっているユビキタスという言葉から説明する。ユビキタスというのはギリシャ語で”偏在する”といった意味を表すものである。ここで私たちが考えているユビキタス環境というのは、小型化したコンピュータが常に私たちを取り囲む環境である。この環境は単にコンピュータが小型化し

て埋め込まれているだけでなく、いつでもどこでも必要な情報にアクセスすることを存在を意識する事なくアクセスできることである。
その環境における機器については、図 5-3-1 に示すように大きく分けて、固定ノードと移動ノードの２種類に分類できる。例えば、建築やそれに付随する機器に埋め込まれてその情報を取り扱う機器は固定されているので固定ノードといえる。PDA (Parsonal Digital Assitant)のような機器に埋め込まれている物は、利用者とともに移動するノード（移動ノード）といえる。また車に埋め込まれた機器などは、機能やサイズなどの条件は固定ノードにに属する要素を持っているが”移動する”という面では、これら両方に属するといえる。

図 5-3-1 ノードの分類

まずこれらノードにの特徴について説明する。

(1) 固定ノード

このノードは場所に”固定”されその環境に特化した情報を扱う。そのため機能の面においても特定の機能に特化していることが多い。例えば、図 5-3-2 に示すような駅のゲートがそれにあたる。

図 5-3-2　駅のゲート

このノードは、駅に入場する際のチケットの有無などによる条件を調べたり、時間の情報から最終電車発車以降は入場を制限するなどの制御を行う。このノードの特徴としては実装面におけるサイズの制約は比較的緩いが、ある特定の処理を行う機能においてはその処理を終了するまでの時間の制限や信頼性を持つ。

(2)　移動ノード

私たちが現在着目しているのは利用者とともに移動し最も直接利用される移動ノードである。このような世界でユビキタス環境とのコミュニケーションを行う移動ノードである汎用情報携帯機器を実現するものが、ユビキタスコミュニケータである。ここでまずコミュニケーションとついて定義しそれを実現したユビキタスコミュニケータと UC-Phone について説明する。

コミュニケーションは以下の３つの概念としてとらえる事ができる。

- 人とのコミュニケーション
  言葉という音声や表情・身振りといった映像を送る事により、相対する人に対して自分の意思疎通を行う。
- モノとのコミュニケーション
  モノにその内容を示すタグ(ucode)を貼付し、この ucode を読み取った内容をキーにサーバを検索しモノの持つ情報をユビキタスコミュニケータに表示する。ユビキタスコミュニケータ自体も ucode として機能し、鍵・お金・チケットといったモノ（電子実態）として機能する。

- 環境とのコミュニケーション
  家電や設備機器に組み込まれたコンピュータやさらには場所につけた ucode との通信を行い、環境の情報を入手したり、環境や情報の操作を行う。

これらに対してユーザに特に負担を強いる事なくコミュニケーションを行える事がユビキタスコミュニケータの目標である。機能として必要な条件を再度整理すると以下である。

- 人とのインタフェース
- ネットワーク通信機能
- ucode 読み取り機能
- 電子実態機能

（イ）ユビキタスコミュニケータの特徴

ユビキタスコミュニケータは T-Engine アーキテクチャをベースとして開発を行っている。T-Engine アーキテクチャとは T-Engine フォーラムと呼ばれる企画推進団体が定義するアーキテクチャである。T-Engine アーキテクチャは CPU ボード・オープンソースなリアルタイムオペレーティングシステム T-Kernel を標準化したものである。図 5-3-3 に T-Engine アーキテクチャにおける標準ボードを示す。

図 5-3-3 T-Engine アーキテクチャによる標準ボード

ユビキタスコミュニケータは、前述したこのアーキテクチャに対してユビキ

タス環境に必須の下記機能を拡張したものである。

a)　人とのインタフェース

b)　ネットワーク通信機能

c)　ucode の読み取り機能

d)　電子実態機能

ユビキタスコミュニケータのハードウエア構成を図 5-3-4 に示す。

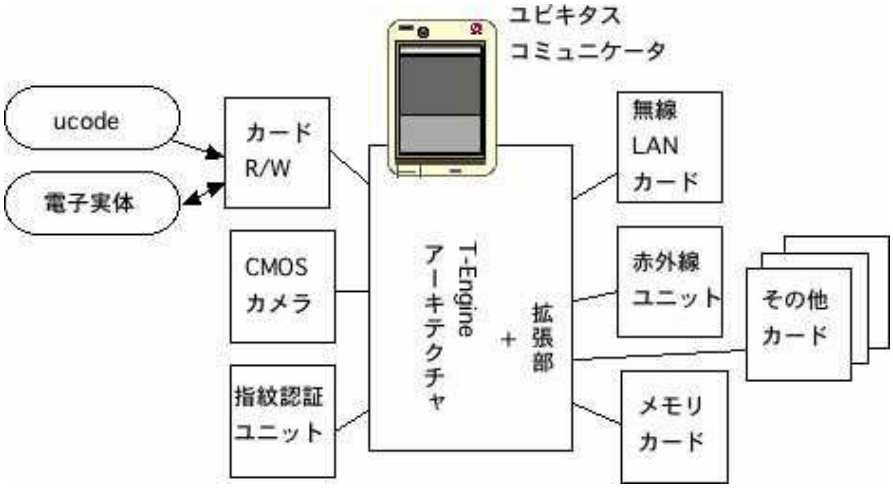


図 5-3-4　ユビキタスコミュニケータの構成

a）人とのインタフェース機能

通常、人とのインタフェースというとウインドウやメニューを LCD などの画面に表示する GUI（Graphical User Interface）をイメージするが、ここではそれに加えて音声による読み上げ機能やタッチパネルなどの機能を持つ

b）ネットワーク通信機能

無線 LAN に限らず、例えばその時点でアクセス可能な赤外線ユニットを利用して通信を試みる機能などである。

c) ucode の読み取り機能

世界のさまざまなモノに RFID やセンサーなどで作られたユビキタス ID タグ（以下 ucode タグ）が埋め込まれる事を想定している。基本的な考え方としては、ucode タグはそのモノに関する情報を格納するが、現在では記憶容量などの制約があるため、すべての情報を格納することはできない。そこで ucode タグにはモノを識別する ID コード（ユビキタス ID）だけを格納し、容量の範囲内で付加的な属性情報を格納している。

図 5-3-5 ucode タグの例

ucode タグに格納できない情報はネットワークの先のデータベースに格納される。ユビキタスコミュニケータは獲得した ucode に応じて情報サービスサーバなどにアクセスして情報サービスを受けることを想定している。ユビキタス環境では実世界中にばらまかれた ucode タグや情報サービスサーバの数が膨大であるため、ucode 解決プロトコルと呼ぶ巨大分散ディレクトリデータベースがこの ucode と情報サービスサーバの対応関係を保持することを予定している。その関係を図 5-3-6 に示す。

図 6　ucode タグと情報サービスサーバとの関係

d) 電子実体（U-Card）機能

デジタル情報は、情報品質を劣化させる事なく完璧に内容を複製・変更できる。このためにチケットなどの有価値情報やアクセス許可のような権利情報をネットワーク経由で配布したり、管理する事には大きなリスクが伴う。電子実体 (U-Card)は、対タンパーチップをベースとして「紙の印刷物」や「金属の鍵」などの物理的な実体を持つ一体性・製造困難性・複製不可能性・改竄困難性・携帯性などの性質を与えた特殊な性質を持ったデジタル情報を実現している。ユビキタスコミュニケータでは、このようなアクセス管理情報を U-Card の「電子実体」としてやり取りすることにより容易にセッティングでき、かつ強固なセキュリティ管理を実現する機能である。

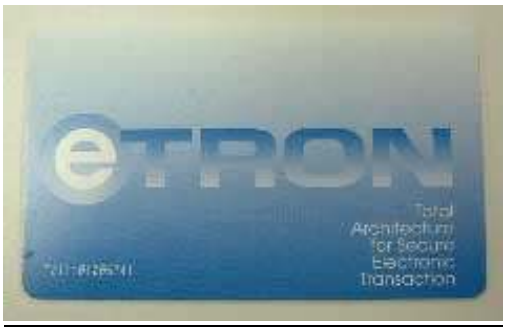


図 5-3-7　U-Card の例

特にこの U-Card と ucode の読み取り機能はパソコンや PDA と呼ばれる小型携帯情報機器などと決定的に異なる機能であり、ユビキタス環境において求められるプライバシに配慮したセキュアな情報環境を生み出すことを可能としている。

図 5-3-8 に示すのはこれら機能を実装したユビキタスコミュニケータの基本モデルである。

図 5-3-8　ユビキタスコミュニケータ（基本モデル）

この基本モデルの仕様を表 1 に示す。

表 1　ユビキタスコミュニケータ（基本モデル）の仕様

機能	仕様
グラフィック	320x320 LCD MPEG-4 動画再生 (QVGA 30fps)

カメラ機能	VGA 15fps
ucode リーダライタ	U-Card チップ　ミューチップ対応
認証	指紋認証ユニット

ウ）ユビキタスコミュニケータの種類

ユビキタスコミュニケータは利用者とともに移動し最も直接利用される移動ノードである。そのため利用形態によりさまざまな種類のユビキタスコミュニケータの実現例が予想され、今回数種類の試作を行った。それら実現例を図 5-3-9 に示す。

図 5-3-9　試作を行ったユビキタスコミュニケータ群

・標準モデル

これは前項で述べたようにすべてのユビキタスコミュニケータの基となるモデルである。

図 5-3-10　基本モデル

タッチパネル付き LCD やカメラにネットワーク通信機能などを持つ。特に他の携帯情報端末と決定的に異なるのは、ucode や情報実態(U-Card)に対する読み取り機器を内蔵している事である。（図 5-3-11）

図 5-3-11　情報実体読み取り装置

表 5-3-2 ユビキタスコミュニケータ（基本モデル）の仕様

機能	仕様
グラフィック	320x320 LCD MPEG-4 動画再生(VGA 30fps)
カメラ機能	VGA 15fps
ucode リーダライ	U-Card チップ　ucode チップ対

タ	応
認証	指紋認証ユニット

・表示強化モデル

標準モデルの情報表示機能を強化したモデルである。タッチパネルつき LCD を基本モデルの QVGA(320x320)から VGA(640x480)とし、一度に表示できる情報を増やしている。

図 5-3-12　表示強化モデル

表 5-3-3 表示強化モデルの仕様

機能	仕様
グラフィック	640x480 LCD MPEG-4 動画再生(VGA 30fps)
カメラ機能	VGA 15fps
ucode リーダライタ	U-Card チップ　ucode チップ対応
認証	指紋認証ユニット

・外部接続強化モデル

表示強化モデルをベースにさらに長時間駆動と外部接続インタフェースを強化したモデル。図 5-3-13 で上側が表示強化モデルで下側のやや厚みが増してコネクタが増えているのが駆動時間強化モデルである。

図 5-3-13　外部接続強化モデル

表 5-3-4 外部接続強化モデル

機能	仕様
グラフィック	640x480 LCD MPEG-4 動画再生(VGA 30fps)
カメラ機能	VGA 15fps
ucode リーダライタ	U-Card チップ　ucode チップ対応
認証	指紋認証ユニット
外部接続端子	シリアル I/O

・バーコード対応モデル

ユビキタスコミュニケータを業務用途（流通など）に強化したモデルである。名称からもわかるように、現在広く普及しているバーコードリーダーを備え情報を入手手段を保持することでレガシーなシステムとの親和性を高めている。

図 5-3-14　バーコード対応モデル

表 5-3- 5　バーコード対応モデル

機能	仕様
グラフィック	640x480 LCD MPEG-4　動画再生（VGA 30fps）
カメラ機能	VGA 15fps
ucode リーダライタ	U-Card チップ　ucode 対応
認証	指紋認証ユニット
入力インタフェース	バーコードリーダー

エ）UC-Phone

ここでは T-Engine アーキテクチャから派生したマイクロ T-Engine アーキテクチャをベースに開発した UC-Phone について説明する。

UC-Phone はその名称からもわかるように主として音声通話によるコミュニケーションを強化したモデルである。

その外観を図 5-3-15 に仕様の概略を表 5-3-6 に示す。

図 5-3-15　UC-phone の外観

表 5-3-6　UC-Phone 仕様概略

通信方式	PHS 方式
表示部	STN 2.6 inch 240x160
RFID リーダ	T junction 2.45GHz
バーコードリーダ	レーザ式　１次元バーコード
外形寸法	143H x 55W x 266D
重量	144g （バッテリ含）
バッテリ容量	690mAh

処理方式の特徴としては以下である。

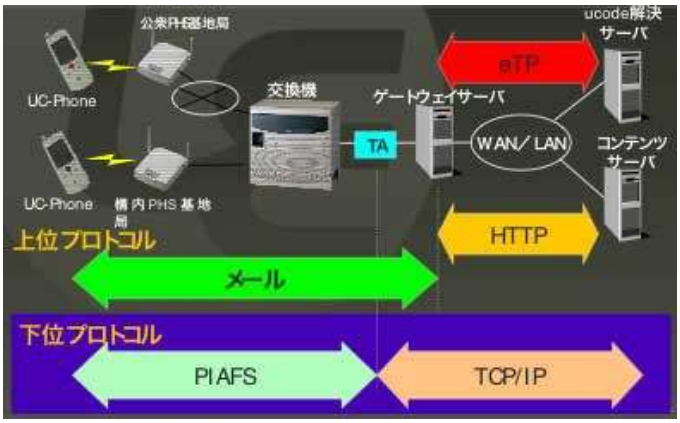


図 5-3-16　UC-phone の特徴

図 5-3-16 に示すように UC-Phone の持つ PHS 機能を利用し、すでに広まっているデータ通信形式である PIAFS 通信を用いて情報通信をおこなう。その際に ucode タグの情報を取り扱う ucode 解決サーバと UC-phone との間にゲートウエイサーバと呼ばれる機能をもつサーバを用意し、ここが ucode 解決などの処理を代理で行う事により UC-phone 側の処理負荷の軽減とすでに広まっている PIAFS 通信プロトコルとの融合をはかっている。

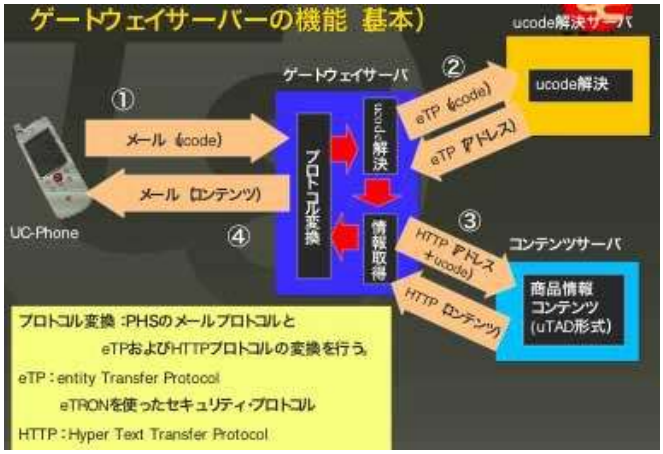


図 5-3-17　ゲートウエイサーバの機能

ゲートウエイサーバと UC-phone を用いた ucode タグからモノの情報を得るまでのながれを図 5-3-17 に示す。

1. UC-phone は複数の ucode（RFID やバーコード）をローカルで読みゲートウエイサーバに送信する。
2. ゲートウエイサーバは、それらを情報サービスサーバでどこにその ucode

に関する情報が存在するのかアドレスを得る。

3. ゲートウエイサーバは得られたアドレスで実際のモノの情報（コンテンツ）を持つサーバーに対して送信を要求する。
4. ゲートウエイサーバは得られたコンテンツを UC-phone に送信する。

5-3-2 コンテキスト情報記述・管理方式

（ア）コンテキスト情報記述管理方式

現在、研究所では uTAD(Ubiquitous TAD)というフレームワークの構築を行っている。これはユビキタスコミュニケータにおいて表示などのコンテンツを扱う定義を行う規格である。ベースとして XML(eXtension Markup Language)をベースとしており、ユビキタスコミュニケータ用のタグを用意している。コンテンツの主たる物は通常のテキストや JPEG などの画像、MPEG4 の動画などであるが、それに対して位置情報の記述方法を検討した。その結果について述べる。

今回は場所に ucode を対応つけることで解決することとした。

図 5-3-18 のように住居表示板に ucode を割当ててある状態を例にとると、住居掲示板にユビキタスコミュニケータをかざすことにより”場所に割り当てられた ucode”をその場所自体から入手することができる。次にそれを ucode 解決サーバに問い合わせる事でその ucode 自体の持つ意味を人がみる事ができるコンテンツとしてコンテンツサーバから得る事ができる。この一連の処理の流れは、今まで実装を行っていたモノに ucode を割当てその情報を得るものと同じ処理の流れである。すなわち”場所自体に ucode を持たせる”ということによりその他のコンテンツと同様の処理で位置情報を管理する事ができる。その結果コンテンツ定義部のヘッダタグに対して ucode と場所との対応の attribute（属性）を記述することで、管理可能になった。

図 5-3-18　ucode と場所との対応付けによる処理

(イ）uTAD コンテンツ記述

ここでは、ucode 解決を行った後、入手するコンテンツ記述について示す。

・XML 要素一覧

XML によるコンテンツの設定を行い、UC の表示部分を制御する為に用意する。XML 要素の一覧を以下に示す。

表 5-3-7　XML 要素一覧

要素名称	内容
ubicontents	すべての要素のルート
contents	画面全体の属性を管理
image	画像を管理
sound	音声を管理
component	タップ制御などを管理
movie	動画を管理

これらの要素を階層的に記述することにより制御を実現するが、処理の高速化を考え読み込む XML ファイルは DTD による検証は行わないなどの制約を設ける。（注：将来の実装に備え現在未使用なタグや属性の記述もある）

・要素の説明

(1) ubicontents 要素

この要素は他のすべての要素のルートとなる。設定する属性は特にない。子供要素の contents 要素は一つのみしか持つことができない

(2) contents 要素

この要素は画面全体に関する属性の管理を行う。設定する属性は以下である。

表 5-3-8 contents 要素

要素名称	属性名称	内容
contents	id	コンテンツ識別子（現在未使用）
	lang	コンテンツの言語識別子
	basepath	読み込みのベースパス
	description	コメント記述
	attribute	コンテンツの属性（現在未使用）

記述例：ベースパスがカレント

〈contens id="" basepath="." description="sample" attribute="" 〉　〈/contents〉

(3) image 要素

この要素は画像に関する属性の管理を行う。設定する属性は以下である。

表 5-3-9　image 要素

要素名称	属性名称	内容
image	url	入力先(2.2 の basepath が存在する場合はそれと合成される)
	pos_x	基準 X 座標
	pos_y	基準 Y 座標
	width	幅
	height	高さ

記述例：入力ファイルが sample.img（幅 480 高 640)で基準座標が(10, 10)である場合

<image url="sample.img" pos_x="10" pos_y="10" width="480" height="640"></image>

(4) sound 要素

この要素は音声に関する属性の管理を行う。設定する属性は以下である。な

お当面は sound 要素に関して複数定義は許さない。

表 5-3-10 sound 要素

要素名称	属性名称	内容
sound	url	入力先(2.2 の basepath が存在する場合はそれと合成される)
	autostart	自動再生

記述例：入力ファイルが sample.wav で自動再生しない場合

<sound url="sample.wav" autostart="false" ></sound>

(5) component 要素

この要素は主として画面タップに関する属性の管理を行う。直接画面には表示されずヒット判定に必要な領域の設定とそのときの動作を記述する。設定する属性は以下であり、複数存在する場合は後書き優先である。

表 5-3-11 component 要素

要素名称	属性名	内容
movie	url	入力先(2.2 の basepath が存在する場合はそれと合成される)
	pos_x	基準 X 座標
	pos_y	基準 Y 座標
	width	幅
	height	高さ
	autostart	自動再生

記述例：基準座標(0,0)として sample.m4v を読み込み自動再生をする場合

<movie url="sample1.m4v" pos_x="100" pos_y="100"

width="480" height=352 autostart="true" ></movie>

（ウ）uTAD 記述例

カレントにある 480x640（UC フルスクリーン）の etron.jpg イメージを貼り付け画面下半分をタップすると次の画面（page2.xml)を読み込む記述例を図 5-3-19 に示す。

図 5-3-19　記述例

5-3-3 位置検出機構

ここではユビキタスコミュニケータにおける位置検出メカニズムについて述べる。すでに前出の”ユーザノードシステムの開発”で示したようにユビキタスコミュニケータは多彩な外部環境からの情報を取り込むことが用件として存在し、とくに移動ノードとしてのユビキタスコミュニケータにおいては、利用者の位置情報の検出とそれを利用した情報処理というのが重要なテーマである。

一概に位置検出といっても様々なものが存在する。すでに実用化され広く認知されているものとしては、GPS（Global Positioning System)がある。これは地球の周りに複数の人工衛星を打ち上げその衛星郡から発せられる電波を受信し、三角測量することで受信者の位置を検出するものである。元々は軍用目的で開発・利用されていたものであるがその後民間にも無料で開放され、今では自動車のナビゲーションシステムや小さい物では携帯電話にまで内蔵され位置を検出できるようになっている。

ただし、この GPS による位置検出はその原理上の誤差が数メートルあるため、人間が感覚的にとらえられる位置誤差とのギャップが大きいことや、カーナビゲーションのように単純な地図との対応関係では、工事や事故などのような情報はリアルタイムで反映することは困難である。

そこで人が違和感を感じない程度の誤差で位置を検出する方法とその位置に対して、実際に人が欲する情報をいかにして関連づけるかということが目標であった。研究試作を行う際に着目したのは、赤外線と RF である。

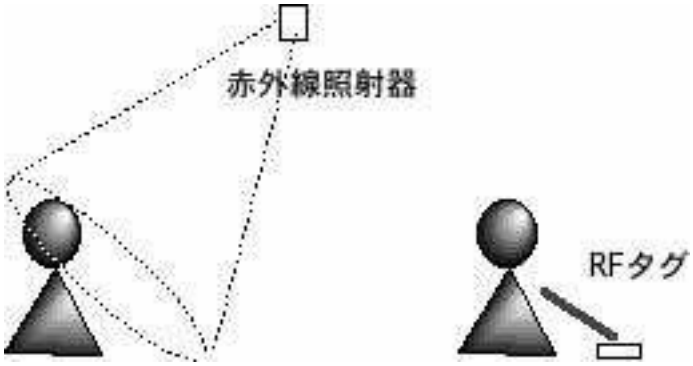


図 5-3-20　位置検出の着目点

赤外線は、家電の世界などで制御リモコンの通信手段として使われているようにすでに信頼性のある技術である。特性として出力を変更する事により到達距離を調整したり、指向性があるので絞り込むことにより赤外線を受信できる範囲を容易に制御する事が可能である。

RF タグに関しては、物理的にセンサをその場所にかざすということからその原理上ピンポイントでの位置を設定するのに適している。また赤外線が干渉するなど使えない場合などに利用することも考えられる。

図 5-3-21 に示すのは今回試作を行った赤外線 ID タグである。

図 5-3-21　赤外線 ID タグ

これは内部でその位置情報をしめす ID を赤外線にのせて発信する機能を有している。その赤外線をユビキタスコミュニケータが持つ機能の図２２に示す部

分で受信し、そこから得られる ID をエミュレーションで ucode と同様の処理をすることで位置情報を割り出すことができる。

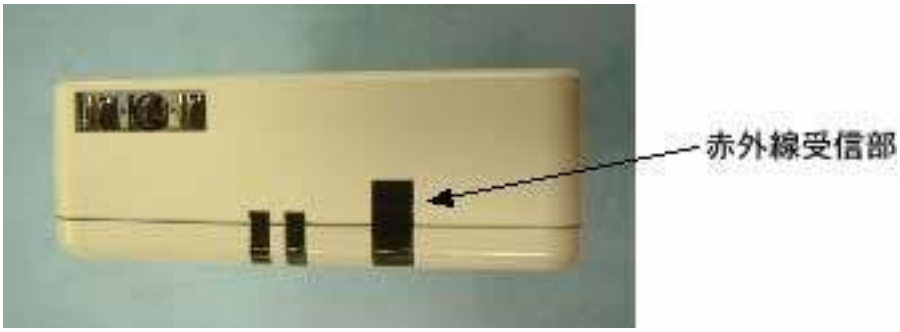


図 5-3-22　赤外線受信部

例えば、赤外線 ID タグをビルの入り口に設置し、登録を行っておく事でそのビルの位置情報を表示したり、さらには工事現場で見かける三角コーンに内蔵することにより工事現場に関する情報をリアルタイムに得る事ができる。（図 5-3-23）

図 5-3-23 赤外線 ID タグによる位置情報の入手

次に RF を利用したタグを用いた位置検出に関して述べる。

図 5-3-24 左側に示すように赤外線と同様に ucode を割り当た RF タグを割当て登録を行う。利用者はここに直接ユビキタスコミュニケータをかざすことで位置情報を入手する。この用途は直接かざすという形を生かした限定された範囲に関する位置情報やなんらかの影響により赤外線タグが使用できないときの代用を想定している。また ucode 利用方式は動作に電源を使用しないためメンテ

ナンスに関する手間やコストがかからないのが利点である。また位置に対して ucode を割り当てることによって、ucode に関する一連の処理（ucode のサーバへの問い合わせ→対応づけられたコンテンツの入手→表示）を利用する事で、位置に関する情報とさらにそれに付加された利用者の求める”位置（場所）の意味する内容”にする情報を得る事ができる。（図 5-3-24 右側）

図 5-3-24　RF タグによる位置情報の検出

また RF タグを衝撃吸収素材の中に埋め込むことで、人が歩く地面などにも直接貼付けることができる。この特性を利用した適用事例を以下に述べる。
今回適用を行ったのは、地面に配置される視覚障害者向け誘導ブロックへの組み込みである。視覚障害者の多くは白杖（身体障害者福祉法における名称では盲人安全杖）を持っている。この白杖の機能は大きく以下の３つである。
1）防御
2）存在を周囲に知らせる
3)情報収集
この中で重要なのは 3)情報収集の機能で、杖を体の前におき周囲の地面などをこの杖で探ることにより障害物などを検出するセンサーの役割である。
しかしこの情報収集方法ではかなりの訓練が必要である事と、得られる情報に限界があり、検出した障害物に対してその内容まで知る事は困難である。
また、移動前にあらかじめルートマップに相当する情報を頭の中に記憶しない

と現在位置を見失いやすく、また途中に障害となる事柄（工事や障害物）などがあると混乱してしまう事例が多く、視覚障害者の移動に関する大きな障害となっている。

そこで誘導ブロックに ucode を割り当てた RF タグを埋め込みそれを利用するシステムを試作する事でこの問題に対する障害を緩和した。視覚障害者向けの施設として普段私たちが目にするのは、歩道などに設置されている黄色い誘導ブロックである。この誘導ブロックは白杖から得られる情報を補足する意味合いでもうけられ、その形状により「とまれ」や「注意」、「進め」という内容を表す。

しかし、実際の視覚障害者の不満は

・バス停や電話ボックスの位置がわからない
・公衆トイレの場所、男女の別、入り口がわからない
・工事中などの危険な場所が分からない

など、その誘導ブロックが「なにに誘導するのか」という意味を表す情報が読み取れない事にある。

これらの問題点にたいして、本研究では図 5-3-25 に示すように誘導ブロックに耐衝撃性機能をもつ ucode を割り当てた RF タグを埋め込こむことからえられる効果で解決をはかった。

図 5-3-25　ucode 付き視覚障害者誘導ブロック

この誘導ブロックを用いた処理の流れを図を使いながら説明する。図 5-3-26 に示すのはインテリジェンス白杖の例で、白杖の先端に ucode を読み出すための

センサを組み込んである。これを用いる事で杖を伝わる手応えで従来の白杖と同じような障害物の感覚を得る事ができながら、かつ地面につけられた ucode を利用者が意識することなく読み出す事ができる。

図 5-3-26　　インテリジェンス白杖

図 5-3-27　　ユビキタスコミュニケータと誘導ブロックを用いた ucode の処理

こうやって読み出した ucode を ucode 解決サーバと呼ばれる情報サービスサーバに送り、その ucode に対応つけられた「位置（場所）が意味する内容」へのアドレスを得る。次にそれをコンテンツサーバと呼ばれる情報の実体を蓄えたサーバに送る事でユビキタスコミュニケータがユーザに提示するのに必要な情報を得る。
またユビキタスコミュニケータにはこのインテリジェンス白杖以外にも様々なセンサを内蔵、もしくは接続する事ができる。そこで図２８のようにユビキタスコミュニケータに地磁気センサをつなぐ。これにより利用者の方向に対する情報を得る事ができるので ucode の情報とあわせると、例えば”右にわき道”といったような現在の向きに対する細かな情報を得る事ができる。これをユビキタスコミュニケータの音声読み上げ機能を使う事により車におけるナビゲーションシステムと同等のサービスを提供する事ができる。

図 5-3-28 地磁気センサーと方角情報と ucode を利用したナビゲーション画面

また最初に述べた赤外線タグもこの地磁気センサーと組み合わせる事でさらに利用者にとってわかりやすい情報を提示する事ができる。たとえば歩道と車道の境目に配置しておけば、間違って車道に出てしまった場合にはその ucode を受け取り、地磁気センサーからの情報を統合して図 5-3-29 に示すように”歩道は左方向です”という具体的な指示をだすことができる。またその処理もユビキタスコミュニケータはリアルタイム OS で動作するためこのような緊急事態にむけた処理を優先的に処理する事ができる。

図 5-3-29　地磁気センサーによる方角情報と
赤外線タグを利用したナビゲーション

5-4 サーバノードシステムの研究開発

5-4-1　ユビキタス PKI

5-1-1 節で報告したしたセキュアチップとセキュアプロトコル eTP との組み合わせで、公開鍵暗号系の処理を IC カードでリアルタイムに実現できる認証方式について、平成 14 年度の成果をベースに 16 ビット版に拡充した。詳細については、5-1-1 節にあわせて記載したので参照されたい。

5-4-2　アドレス解決サーバ

（ア）　目的

ユビキタスネットワーキングの基盤技術として、あらゆるモノに貼り付けられたセキュアＩＣチップに付与されるＩＤによるアドレス解決を実現することを目的として、ＩＣチップに付与されるＩＤとＩＰアドレスの組を管理するＩＤ解決サーバを開発する。

（イ）　アドレス解決サーバの概要

ユビキタスネットワークでは、実世界のさまざまなモノに、ＲＦＩＤやセンサーなどでつくられたＩＣチップが埋め込まれる。基本的な考え方としては、ＩＣチップはそのモノに関する情報を格納するが、現在では記憶容量等の制約があるため、すべての情報を格納することはできない。そこで、ＩＣチップにはモノを識別する識別子コードだけを格納し、容量の範囲内で付加的な属性情報を格納している。ＩＣチップに格納できない情報は、ネットワークの先のデータベースに格納される。

このＩＣチップからＩＤを獲得する携帯情報端末は、獲得したＩＤに応じて情報サービスサーバなどにアクセスして情報サービスを受ける。ユビキタスコンピューティング環境では、 実世界中にばら撒かれたＩＣチップや情報サービスサーバの数が膨大であるため、アドレス解決サーバが持つ巨大分散ディレクトリデータベースが、このＩＣチップの識別子と情報サービスサーバの対応関係を保持する。これがＩＣチップに格納された情報があらわす現実世界と、情報サーバ上の仮想世界との間の橋渡しをする重要な基盤システムである（図5-4-1）。

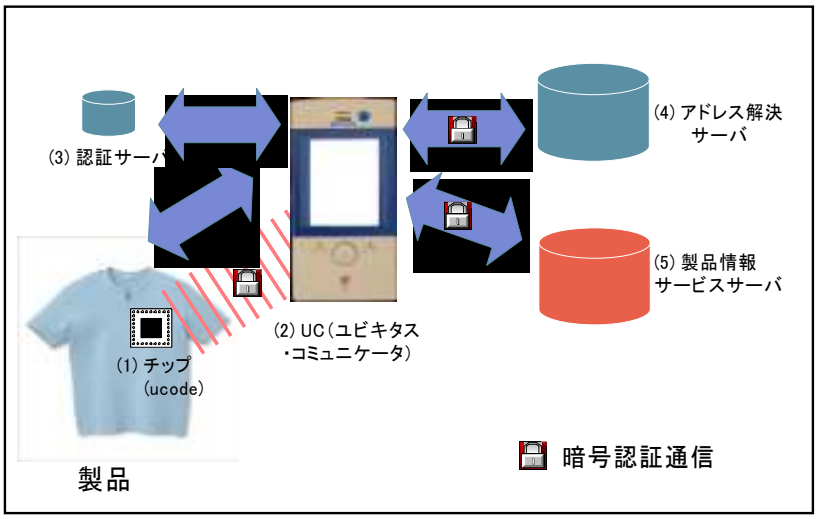


図 5-4-1 アドレス解決の仕組み

（ウ）　システム構成

・ハードウェア・ネットワーク構成

アドレス解決サーバを設置するユビキタス ID 関連システムのハードウェア及びネットワーク構成を以下に示す。

アドレス解決サーバは、アドレス解決のリクエストを受け付けるアドレス解決サーバ本体、およびアドレスデータを格納し必要に応じて解決結果をアドレス解決サーバに返却するアドレス DB サーバから構成される。

アドレス DB サーバはファイアウォールで守られたネットワークセグメントに位置し、ユビキタスコミュニケータ等からのアドレス解決のリクエストはアドレス解決サーバのみが受け付けることでアドレスデータのセキュリティ強度を高めている。

・ソフトウェア構成

アドレス解決サーバのソフトウェア構成を以下に示す。

アドレス解決サーバのモジュールは、セキュア通信を実現するセキュア発行モジュールが動作するサーバ上で動作する構成とした。その他、Linux、Java、PostgreSQL といったオープン系プラットフォーム上での動作を前提として開発した。

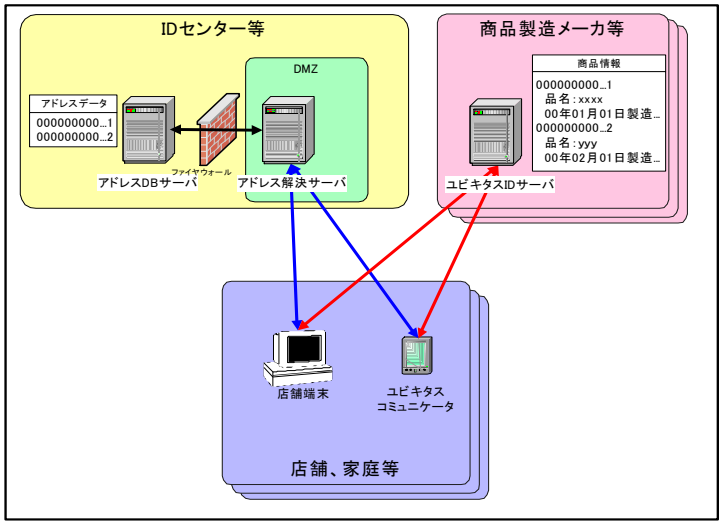


図 5-4-2　ハードウェア・ネットワーク構成図

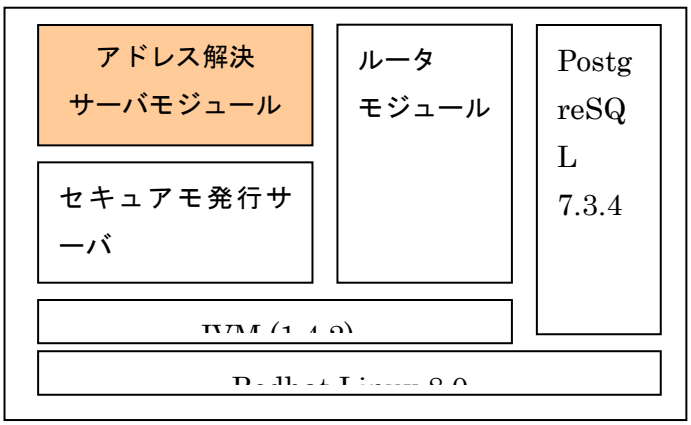
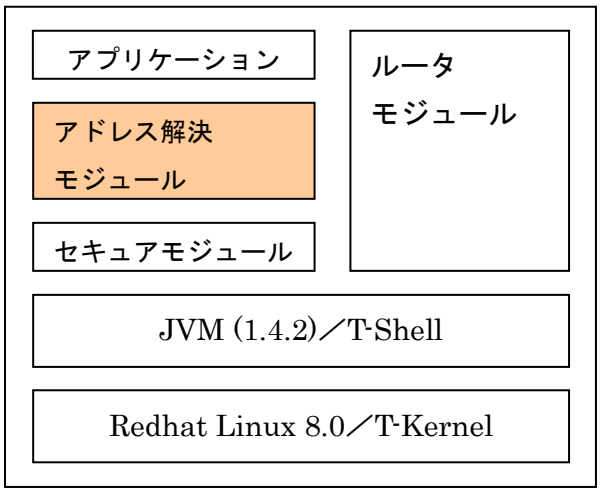


図 5-4-3 ソフトウェア構成(uID 解決サーバ)

また、同時にアドレス解決サーバへアドレス解決リクエストを実行するためのアドレス解決クライアントモジュールを開発した。

図 5-4-4 ソフトウェア構成(uID 解決サーバ)



（エ） 開発の成果

開発済みのアドレス解決サーバモジュールは、ユビキタスＩＤサーバ等を統合的に活用したユビキタス情報提供システムに組み込まれ、使用された。

その他評価等についてはユビキタスＩＤサーバの項で総合的に述べることとする。

5-4-3 セキュア発行サーバ

（ア） 目的

ユビキタスネットワーキングの基盤技術として、あらゆるモノに貼り付けられたＩＣチップに付与されるＩＤが示すアドレス情報、及びアドレスが示す格納先に存在する情報等の電子的価値情報をセキュアに発行することを目的として、セキュア発行サーバを開発した。

（イ） セキュア発行サーバの概要

セキュア発行サーバは、ユビキタスコンピューティング環境における有用な電子情報を提供する全てのサーバノードに求められる機能である。

すべてのモノに ID がふられ、様々な属性情報が飛び交うユビキタス社会においては、情報の取得がより容易になるとともにそれらの盗聴・改竄等といった危険性も多くはらんでいる。このような状況では、モノの情報（コンテンツ）はもとより、そのコンテンツの格納先を示すアドレス情報、アドレス解決を要求するリクエスト情報等に関しても十分なセキュリティ対策を施す必要がある。

セキュア発行サーバは、これらの電子的価値情報を安全に配送するための通信基盤である。具体的には、情報取得を要求するクライアントと情報を提供するサーバ双方にバーチャルプライベートネットワークを構築するセキュア発行サーバモジュールを導入し、セキュア通信を実現する。

また、コンテンツデータだけでなく、アドレス解決を要求するリクエスト送信、レスポンス取得等にもセキュア通信を採用している。

（ウ） セキュア発行サーバのシステム構成

・ハードウェア・ネットワーク構成

セキュア発行サーバの実現方法として、前年度開発済みの「リアルタイムＰＫＩを用いたセキュア基盤ネットワーク」を流用する。

なお、アドレス解決リクエスト／レスポンスなど、より即時性が求められる利用形態を鑑み、処理時間のかかる電子的価値情報の生成、転送機能については採用せず、セキュアＩＣチップを用いた鍵生成とそれを用いた暗号認証通信路の構築によるバーチャルプライベートネットワークを確立することでセキュリティを確保することとした。

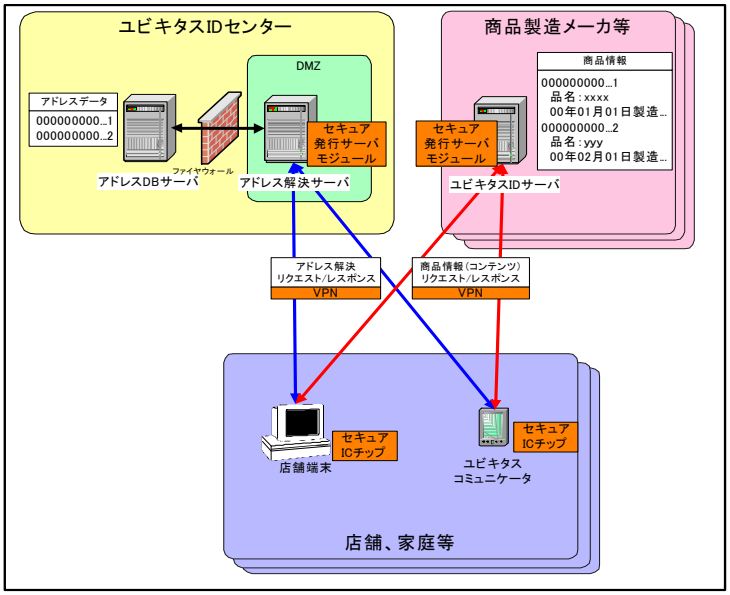


図 5-4-5 ハードウェア・ネットワーク構成図

・ソフトウェア構成

セキュア発行サーバのソフトウェア構成を以下に示す。

セキュア発行サーバのモジュールは、アドレス解決サーバモジュールや商品情報サーバモジュールといったアプリケーションサーバモジュールの下位に位置し、これらのサーバが扱うデータをセキュアに提供するためのミドルウェアとして動作する。

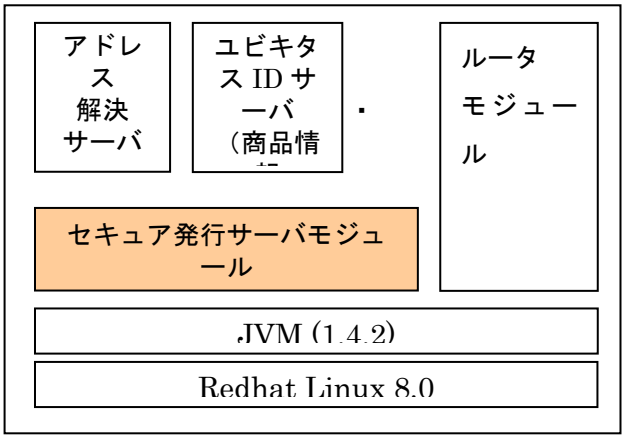


図 5-4-6 ソフトウェア構成(uID 解決サーバ)

また、同時にセキュア発行サーバモジュール、セキュア IC チップと連動し、暗号認証通信を実現するクライアント側セキュアモジュールを開発した。

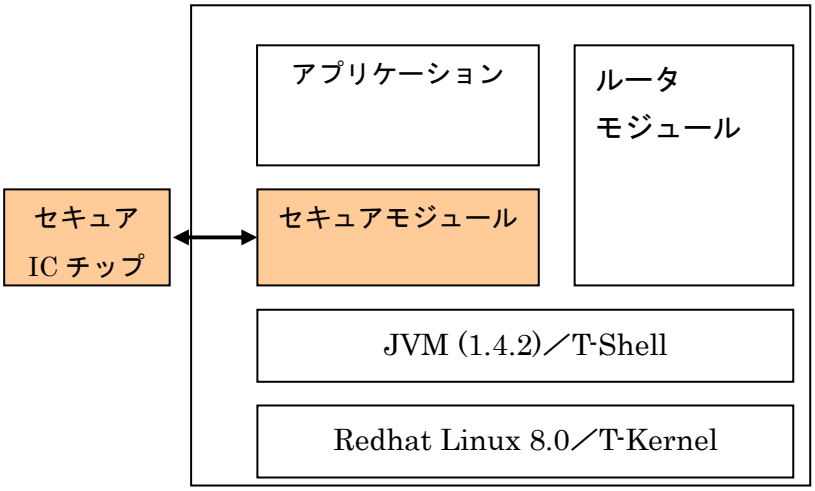


図 5-4-7 ソフトウェア構成(uID 解決サーバ)

（エ） 開発の成果

開発済みのセキュア発行サーバモジュールは、ユビキタスＩＤサーバ等を統合的に活用したユビキタス情報提供システムに組み込まれ、使用された。

その他評価等についてはユビキタスＩＤサーバの項にて総合的に述べることとする。

5-4-4 ユビキタス ID サーバ

（ア） 目的

ユビキタスネットワーキングの基盤技術として、あらゆるモノに貼り付けられたＩＣチップに付与されるユビキタスＩＤとそのＩＤが付与されたオブジェクトに関する情報を提供するサーバの名前やアドレス対応関係を管理し、ユビキタスＩＤ解決プロトコルに従った手順でユーザノード等に情報提供するといったサービスをバックエンドで支えることを目的として、ユビキタスＩＤサーバを開発する。

（イ）　ユビキタスＩＤサーバの概要

あらゆるモノにＩＤがふられ、それらのモノがネットワークを介して接続されたユビキタス社会においては、まずそのモノが「何であるか」「どのようなモノであるか」といった情報を迅速かつ正確に知ることが重要であると考えられる。このような分野で近年注目されているのが、ＢＳＥ問題等からに関心が高まっている食品のトレーサビリティサービスである。生産過程における履歴（食肉用家畜の育成情報、青果物の生産時の農薬使用状況など）、流通過程での入出荷日時の履歴、搬送時の温度、販売店の入荷履歴、加工状況といった、その食品に関わる情報（トレース情報）は消費者の大きな関心事となっており、実フィールドからの有益なフィードバックも期待できる。
このような背景から、ユビキタスＩＤサーバの応用モデルとして食品トレーサビリティを採用し、それを具体的に実現するユビキタス情報提供システムを開発することとした。

対象商品は、食品の中でも比較的流通形態が単純でモデル化しやすいことから、青果物を取り上げることとした。

（ウ）　想定モデル

本システムの開発にあたり、想定した食品トレーサビリティモデルは以下の通りである。

品目	生産段階	流通段階	販売段階	消費段階
青果物（品物に対して生産者、生産ロットが特定できる	肥料、農薬の散布、作業状況などを現場で各種条件とともにチェックし、同時に生産状況に	生産者からの出荷時間、集荷場の入出荷時間および店舗への到着時	店舗では、生産段階で、どのように生産されたかという過程および流通過程が、トレースされた	家庭では、消費者用端末により、いつでもトレース情報を閲覧できる。それにより、

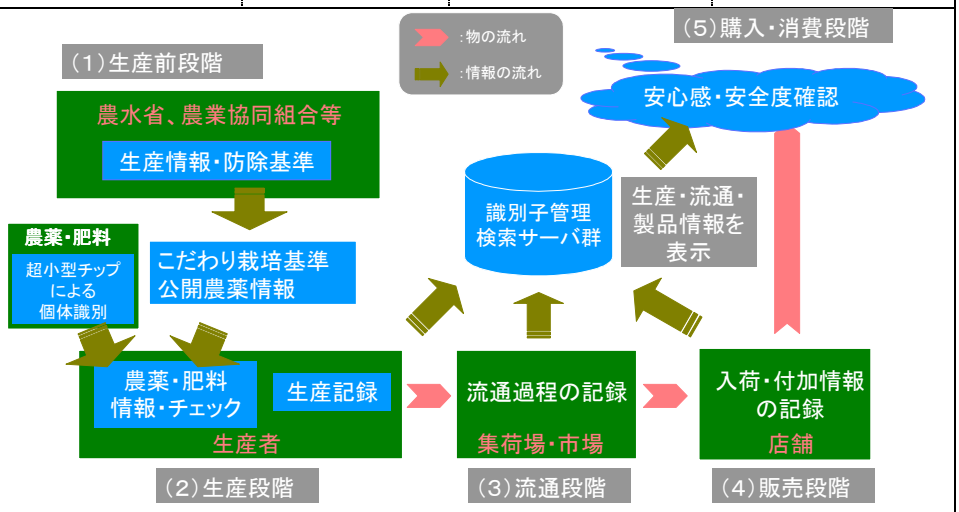
<table>
<tr><td rowspan="2">個選）</td><td>関してトレースのための記録を行う。</td><td>間を記録する。</td><td>情報に基づき、店舗端末によって表示される。</td><td>消費者は、自身が購入した食品の品質、安全性をチェックできる。</td></tr>
<tr><td colspan="4"> 
</td></tr>
</table>

図 5-4-8 食品トレーサビリティモデル

（エ）　システム構成

想定した食品トレーサビリティモデルを元にユビキタス情報提供システムのシステム化検討を行った結果、以下の４つのシステムが統合された構成を採用した。なお、アドレス解決、ユビキタスＩＤの各サーバにはセキュア発行サーバモジュールをミドルウェアとして使用しセキュリティを確保した。

●アドレス解決システム：青果物等に添付されたＩＣチップのＩＤから、生産／流通の情報を引き出すためのアドレス解決サービスを提供する。

●ユビキタスＩＤシステム：青果物の生産／流通に関わる情報を提供するシステム。

●生産／流通システム：青果物の生産履歴等のトレース情報を取得し、入出荷等の流通履歴等とともにユビキタスＩＤシステムに登録する。

●店舗システム：販売店舗に入荷した青果物の入荷日時等の流通情報を登録する。また、店舗の売場、家庭等でＩＣチップを読み取り、アドレス解決後、ユビキタスＩＤシステムから商品情報を取得し表示する。

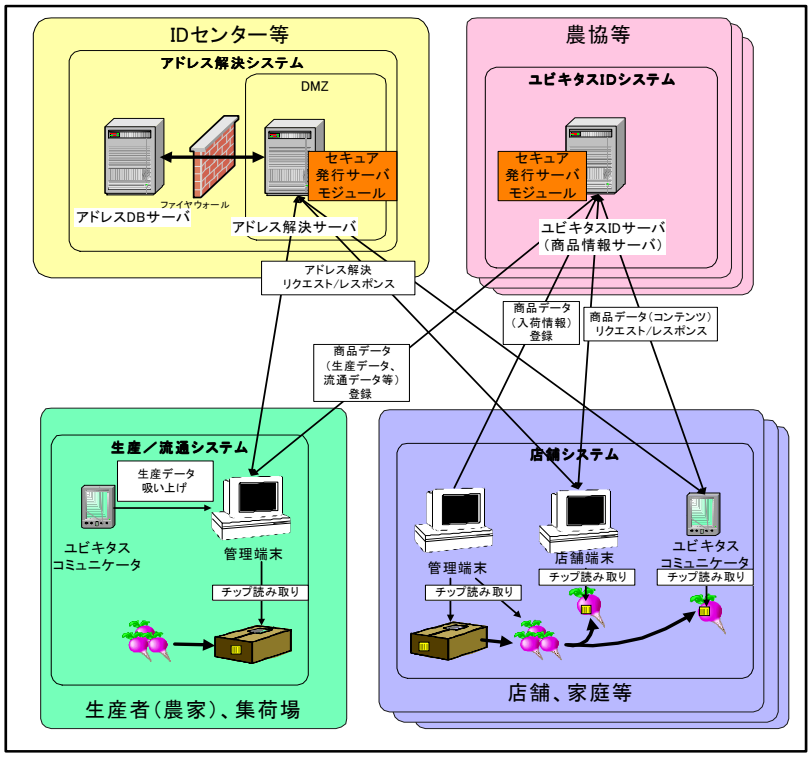


図 5-4-9　ユビキタス情報提供システム構成図

(エ)　システムイメージ

システムの実現イメージを以下に示す。

●　店舗端末とＩＣチップ付き青果物

図 5-4-10 店舗端末と IC チップつき青果物

●商品情報提供例

図 5-4-11 商品情報提供例

表 5-4-1. 提供情報一覧

生産物情報	品名、品種、品質、等級、成分、栽培方法
生産地情報	生産者名、生産地、土壌の説明、生産地・生産者の写真
生産履歴情報	肥料散布時期、農薬散布時期、種蒔き・収穫時期 ※ 別画面で「神奈川県環境保全型農業」を説明した ※ 別画面で使用農薬詳細情報(農薬使用基準を対比させて表示)を説明した
流通情報	入出日付(農協側)、入荷日付(ストア)
フリー情報	生産者からのコメント ※ 生産者が作物にこめた思い、レシピなど 店舗からのコメント ※ 商品鮮度の紹介、味の説明など、店舗の創意工夫部分

●ユビキタスコミュニケータでの表示例

図 5-4-12　表示例

(オ)　開発の成果

　ユビキタスＩＤサーバのほか、アドレス解決サーバ、セキュア発行サーバを含むユビキタス情報提供システムを活用し、青果物の生産、流通、販売の全ルートをカバーしたフィールド試験を行い、有効性検証を行った。

●フィールド試験の概要

・実証スケジュール

平成 15 年 7 月～10 月：事前検討

平成 15 年 11 月～平成 16 年 2 月：生産段階での実証

平成 16 年 1 月 7 日～平成 16 年 2 月 6 日：流通段階での実証

平成 16 年 1 月 8 日～平成 16 年 2 月 6 日：店舗段階での実証

・実証内容

【生産段階】

実施場所：よこすか葉山農協　長井地区組合員圃場

実施規模：８農家８圃場

実施内容：生産システムおよびＩＣチップを用いた青果物の生産情報の自動取得

【流通・販売段階】

実施場所：よこすか葉山農協　長井支店集荷場

京急ストア　平和島店

京急ストア　能見台店

京急ストア　久里浜店

想定規模：出荷用段ボール数　約 2,300 個

出荷作物数　約 20,000 個

出荷用タグ数　約 2,300 個

商品用タグ数　約 25,000 個

実施内容：

- 流通システム、店舗システムおよびＩＣチップを用いた入出荷の支援および入出荷情報の記録
- 店舗システムを用いた青果物へのＩＣチップ貼付

・ 店舗端末（各店舗１台）による一般消費者へのＩＣチップ付き青果物の商品情報の開示および販売
・ ユビキタスコミュニケータ（各店舗３ずつ台配布）を用いた消費者モニタによる家庭での商品情報の閲覧

●フィールド試験結果

【生産／流通段階利用状況】

期間：平成１６年１月７日～２月５日
出荷回数：２４回（期間中の休市日を除く毎日）
出荷箱用ＩＣチップ数：２，３０１個

【店舗端末、ユビキタスコミュニケータ利用状況】

期間：平成１６年１月８日～２月６日
商品用ＩＣチップ数：２４、９２７個
商品情報アクセス件数：５，８５５件

●フィールド試験による有効性の検証

・生産／流通過程の商品情報の登録

出荷箱用ＩＣチップ、青果物用ＩＣチップを読み取り、アドレス解決サーバにてアドレス解決し、指定されたアドレスに存在するユビキタスＩＤサーバへ商品情報を登録する一連の過程において、生産現場（農協）、店舗とも致命的な運用遅延等は発生せず、スムーズな入出荷、登録作業が行われた。

また、従来伝票など紙でやりとりされていた入出荷処理がペーパーレス化され、より作業効率が高まるという副次効果も生まれた。

・商品情報表示内容

ユビキタス情報提供システムで提供される商品情報の内容については、アンケート結果より概ね満足との回答を消費者モニタより得た。

・商品情報表示速度

店舗端末における商品情報表示については、ＡＤＳＬまたは光回線等、高速な回線を用いてユビキタスＩＤサーバ、アドレス解決サーバへ接続しており、ＩＣチップを読み取ってから約１０秒以内と満足のいく応答時間であった。ユビキタスコミュニケータを使用した商品情報表示については、ＰＨＳを使用したことから、ＰＰＰによる回線接続の確立、およびアクセスライン自体の低スピード（１２８ｋｂｐｓ）もあり、約３０秒程度と必ずしも満足なレスポンスタイムではなかった。

●今後の課題

・表示速度の改善

ユビキタスコミュニケータを使用した商品情報表示でのレスポンスタイムについては、改善の必要性がある。今回採用したＰＨＳ回線の性能、性質にも大きく依存するが、セキュア発行サーバモジュール、セキュアＩＣ　チップによる暗号認証通信の確立に起因した遅延も考えられる。

あらゆる場面でネットワークとコンピュータを利用することを想定したユビキタスコンピューティング社会においては、ユビキタスコミュニケータのような小型で比較的性能の低い機器でもセキュリティとリアルタイム性を要求されることが予想されるため、ネットワークインフラの種別によらず、その場面にあった効率的で必要十分なセキュア通信を実現するのが肝要である。

こうした理由から、セキュア発行サーバ等による暗号認証通信についても通信内容／処理内容の解析を進め、ボトルネックの改善等、パフォーマンスチューニングを継続して進める必要があるだろう。

・セキュリティの評価、適正化

セキュア発行サーバモジュール等による暗号認証通信路の採用、ユビキタスＩＤセンター内にサーバ機を設置するなどにより、今回のような期間・規模のフィールド試験では、特にセキュリティ上の問題が発生することはなかった。

今後は、同様のシステムアーキテクチャを採用し、多数のフィールド試験、ひいては実用事例が発生することが想定されるため、現状のシステムアーキテクチャによる十分なセキュリティ診断、評価を実施し、運用面も含めたセキュリティ強度の適正化を目指す必要がある。

5-5 システム統合技術の研究開発

5-5-1 次世代通信プロトコルと既存ネットワークプロトコルとの相互接続技術

（ア）　目標

ソフトウェアを暗号化して配布し、ソフトウェアを実行する際にライセンスチケットを利用してソフトウェアの復号・課金を行うことにより、ソフトウェアの不正利用を防止しながらソフトウェアの流通促進を図るシステムの構築。

（イ）　成果概要

総合的に最適なシステム統合方式が選択可能な基盤を実現するために、ユビキタス環境におけるソフトウェア等のコンテンツの流通を促進、ライセンス制御可能なコンテンツ配布流通用ソフトウェアを設計し、その一部を試作した。

（ウ） 成果の詳細

(1) 流通コンテンツの範囲

本システムで流通させるコンテンツは、ソフトウェアおよび静的コンテンツの2種類を想定している。本システムで流通させるターゲットとしては、下記の通りである。

- ソフトウェアの種類
  - ✧ アプリケーション
  - ✧ サブシステム
  - ✧ デバイスドライバ
  - ✧ 動的ライブラリ
- コンテンツの種類
  - ✧ 静的ライブラリ
  - ✧ 音楽・映像ソフト
  - ✧ 電子ブック

(2) システム機能概要

下記機能について今年度設計を行った。以降に、これら機能について説明を行う。なお、今年度は、上記に示す流通コンテンツのうち、サブシステムおよびデバイスドライバをターゲットとしている。

- 暗号化ソフトウェアの生成機能
- ライセンスチケットを利用したライセンス制御機能

(2-1) 用語定義

本システム機能概要で利用している用語について定義する。

(2-1-1) ライセンスチケット

ライセンスチケットは、当研究所で開発を行ったセキュアデータキャリアチップ内に格納する、暗号鍵及び暗号鍵を使った価値情報操作に必要な情報をまとめたデータ形式であり、ソフトウェアの復号・課金を行うために利用するものである。以下に、ライセンスチケットの形式について示す。

図 5-5-1　ライセンスチケットの形式

表 5-5-1 ライセンスチケットの項目

項目名	説明	length	項目名
ライセンスチケット ID	ライセンスチケットの ID	2byte	セキュリティチップ内で一意となる ID
暗号方式指定子	KEY(0)及び KEY(1)が、どの暗号アルゴリズムの鍵であるかを指定する。	2byte	0x01　DES(KEY(0)のみを使用) 0x02　3DES 0x03　Rijndael 0x04　Hierocrypt-3 0x05　Camellia
KEY(0)～(1)	暗号鍵	16byte	暗号化する際に利用する鍵
BANK ID	銀行 ID	2byte	
支払残高	ライセンスチケットの残高	4byte	単位はライセンスチケットによって任意に設定可能
LEN	付帯情報のオクテット数	4byte	LEN
付帯情報	その他付帯情報	～ 216byte	

(2-1-2) 標準コンテンツ形式

標準コンテンツ形式は、当研究所で開発を行ったセキュアデータキャリアチップ内で価値情報操作を行うことができるコンテンツデータ形式である。セキュリティチップ内のライセンスチケットを操作することによって、標準コンテンツ形式内のデータを暗号化／復号化すると同時に、ライセンスチケット内部の課金情報をアトミックに変更することができる。この仕組みにより、暗号化されたコンテンツを復号する時に確実に課金することが可能となる。

(a) 標準コンテンツ形式（ライセンスチケット内の鍵による暗号化を実行する前）

図 5-5-2　標準コンテンツ形式(暗号化前)

表 5-5-2 標準コンテンツ形式(暗号化前)の項目

項目名	説明	length
コンテンツヘッダ	コンテンツを識別するために必要なデータ	8byte
課金ヘッダ	課金を行う際に必要となるデータを格納するエリア	8byte
暗号方式指定子	元の KEY(0)及び元の KEY(1)が、どの暗号アルゴリズムの鍵であるかを指定する。	2byte
LEN	コンテンツ本体のオクテット数	2byte
元の KEY(0)～(1)	最大 16 バイトの暗号鍵	16byte
コンテンツ本体(0)～(m)	ここでは、ソフトウェア暗号鍵を格納する	-
署名トレーラ	上記データに対する署名を格納するエリア	-

標準コンテンツ形式をライセンスチケット内に格納されている鍵を利用して暗号化すると、標準コンテンツ形式（暗号化後)、暗号化された標準コンテンツ形式をライセンスチケットの鍵を利用して復号化すると標準コンテンツ形式（復号化後）になる。以降、それぞれのデータ形式について示す。

(b) 標準コンテンツ形式（ライセンスチケット内の鍵による暗号化後）

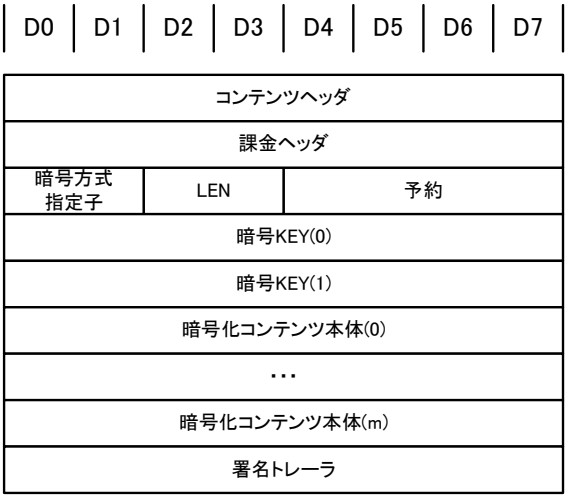


図 5-5-3　標準コンテンツ形式(暗号化後)

表 5-5-3 標準コンテンツ形式(暗号化後)の項目

項目名	説明	length
暗号 KEY(0)～(1)	ライセンスチケットの鍵によって暗号化された元の KEY(0)～(1)	16byte
暗号コンテンツ本体(0)～(m)	元の KEY(0)～(1)によって暗号化されたコンテンツ本体(0)～(m) 本方式の場合は、元の KEY で暗号化されたソフトウェア暗号鍵が格納される	-

(c) 標準コンテンツ形式（ライセンスチケット内の鍵による復号化後）

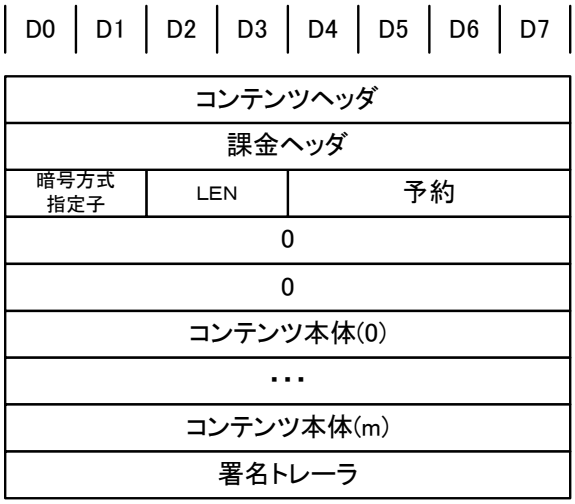


図 5-5-4　標準コンテンツ形式(復号化後)

表 5-5-4 標準コンテンツ形式(復号化後)の項目

項目名	説明	length
コンテンツ本体(0)～(m)	復号化されたコンテンツ本体(0)～(m) 本方式の場合は、復号されたソフトウェア暗号鍵	16byte

	が格納される	

(2-2) サーバによる暗号化ソフトウェアの生成機能

本機能は、ソフトウェア本体、課金額、ライセンスチケット指定子、およびその他データを入力として受け付け、計算処理を行い、配布可能な暗号化ソフトウェアを出力するものである。

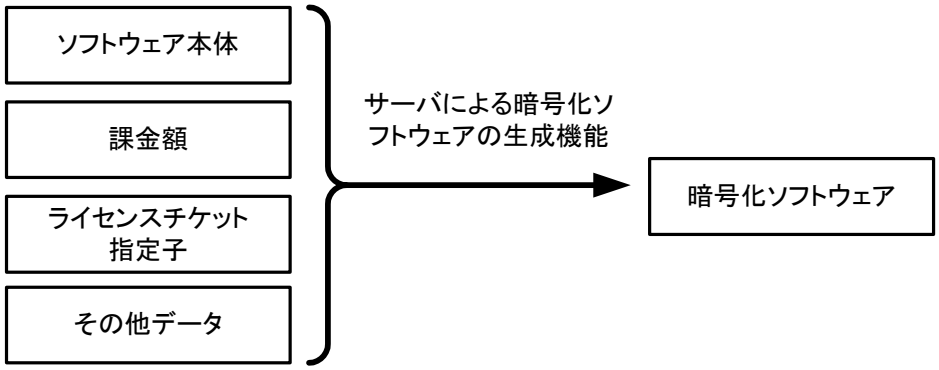


図 5-5-5　サーバによる暗号化ソフトウェアの生成

本機能は、さらに以下の各機能から構成される。

- 暗号化ソフトウェア生成受付機能
- ソフトウェア暗号鍵生成機能
- ソフトウェア暗号鍵によるソフトウェア暗号化機能
- 元の KEY 生成機能
- コンテンツ本体生成機能
- 標準コンテンツデータ生成機能
- 標準コンテンツデータ暗号化機能
- 暗号化ソフトウェア結合機能

これらの各機能について、詳細を以下に記す。

(2-2-1) 暗号化ソフトウェア生成受付機能

(a) 機能概要

以下のデータを暗号化ソフトウェアの生成要求を受け付ける。

- ソフトウェア本体
- 課金額

- ライセンスチケット指定
- その他データ

(b) 機能補足説明

暗号化ソフトウェア生成受付機能に関する補足説明を以下に示す。

(b-1) ソフトウェア本体について

- 研究所にて開発したリアルタイムカーネル上で動作する、ローダにより実行可能な ELF[1]形式のソフトウェア

(b-2) 課金額について

- ライセンスチケット利用時（ソフトウェア実行時）に課金される額
- 不正な額が課金されることを防ぐため、ベンダより提供された正しい数値データであること
- 例えば、額に対して付与されている署名を検証する等の方法が考えられる
- 指定されない場合は、課金額は 0 として処理されること

(b-3) ライセンスチケット指定子について

- 課金するライセンスチケットが格納されているセキュリティチップの ID およびライセンスチケット ID

(b-4) その他データ

- その他、暗号化ソフトウェアを生成する上で必要となるデータ

(2-2-2) ソフトウェア暗号鍵生成機能

(a) 機能概要

ソフトウェア本体を暗号化する暗号アルゴリズムを決定するとともに、暗号化する際に利用される鍵（以下、「ソフトウェア暗号鍵」と呼ぶ）を生成する。

(b) 機能補足説明

ソフトウェア暗号鍵生成機能に関する補足説明を以下に示す。

(b-1) 暗号アルゴリズムについて

暗号アルゴリズムは下記方式のいずれかであること

- DES
- 3DES
- Camellia
- Rijndael

[1] Executable and Linkable Format

・ Hierocrypt-3

(b-2) 暗号鍵について

暗号アルゴリズムに合致した鍵を生成すること

・ DES であれば 64bit、それ以外であれば 128bit

(2-2-3) ソフトウェア暗号鍵によるソフトウェア暗号化機能

(a) 機能概要

ソフトウェア本体を上記ソフトウェア暗号鍵生成機能にて生成されたソフトウェア暗号鍵およびアルゴリズムを用いて暗号化する。

(b) 機能補足説明

ソフトウェア暗号鍵によるソフトウェア暗号化機能に関する補足説明を以下に示す。

(b-1) 暗号鍵について

ソフトウェア暗号鍵生成機能で生成した暗号鍵を利用する。

(b-2) 暗号アルゴリズムについて

ソフトウェア暗号鍵生成機能で決定した暗号アルゴリズムを利用する。

(b-3) 暗号化部分について

ソフトウェア全体を暗号化する。

(2-2-4) 元の KEY 生成機能

(a) 機能概要

標準コンテンツデータ内のコンテンツ本体を暗号化する暗号アルゴリズムを決定するとともに、標準コンテンツ（ここではソフトウェア暗号鍵等）の暗号化の際に利用する鍵（以下、「元の KEY」と呼ぶ）を生成する。

(b) 機能補足説明

元の KEY 生成機能に関する補足説明を以下に示す。

(b-1) 暗号アルゴリズムについて

暗号アルゴリズムは下記の共通鍵暗号方式のいずれかであること

・ DES
・ 3DES
・ Camellia
・ Rijndael
・ Hierocrypt-3

(b-2) 暗号鍵について

暗号アルゴリズムに合致した鍵を生成すること

(2-2-5) コンテンツ本体生成機能

(a) 機能概要

標準コンテンツデータ内に格納するコンテンツ本体を生成する。コンテンツ本体は、下記で構成されている。

- ソフトウェア暗号鍵、ソフトウェア暗号化アルゴリズム、ソフトウェア暗号鍵 length

(b) 機能補足説明

コンテンツ本体生成機能に関する補足説明を以下に示す。

(b-1) ソフトウェア暗号鍵について

ソフトウェア暗号鍵は、ソフトウェア暗号鍵生成機能にて生成されたソフトウェア暗号鍵であること

(b-2) ソフトウェア暗号化アルゴリズムについて

ソフトウェア暗号化アルゴリズムは、ソフトウェア暗号鍵生成機能にて指定されたアルゴリズムであること

(b-3) ソフトウェア暗号鍵 length について

ソフトウェア暗号鍵のオクテット数を表していること

(b-4) コンテンツ本体について

上記で説明したソフトウェア暗号鍵、ソフトウェア暗号化アルゴリズム、および length を連結したもの

必要に応じて、パディング処理を行う

(2-2-6) 標準コンテンツデータ生成機能

(a) 機能概要

ソフトウェアを実行する際に利用する標準コンテンツデータ形式のデータを生成する。

(b) 機能補足説明

標準コンテンツデータ生成機能に関する補足説明を以下に示す。

(b-1) コンテンツヘッダについて

コンテンツ ID 等を格納する

(b-2) 課金ヘッダについて

課金額等を格納する

(b-3) 暗号方式指定子および、元の KEY について

元の KEY 生成機能にて生成された暗号方式指定子、および元の Key を格納する

(b-4) コンテンツ本体について

コンテンツ本体生成機能にて生成されたコンテンツ本体データ

(2-2-7) 標準コンテンツデータ暗号化機能

(a) 機能概要

標準コンテンツデータ生成機能にて生成したコンテンツデータに対する暗号化をライセンスチケットの鍵を利用して暗号化し、暗号化コンテンツデータを生成する。

(b) 機能補足説明

標準コンテンツデータ暗号化機能に関する補足説明を以下に示す。

(b-1) 暗号化する際に必要なデータについて

- 標準コンテンツデータ生成機能で生成した標準コンテンツデータ形式のデータ
- 暗号化する際に利用するライセンスチケットの発行先セキュリティチップ ID およびライセンスチケット ID
- セキュリティチップ ID およびライセンスチケット ID による検索の結果、課金対象とするライセンスチケットが存在しない場合は、エラーを返却する。

(b-2) 暗号化について

暗号化の依存関係について下図に示す。

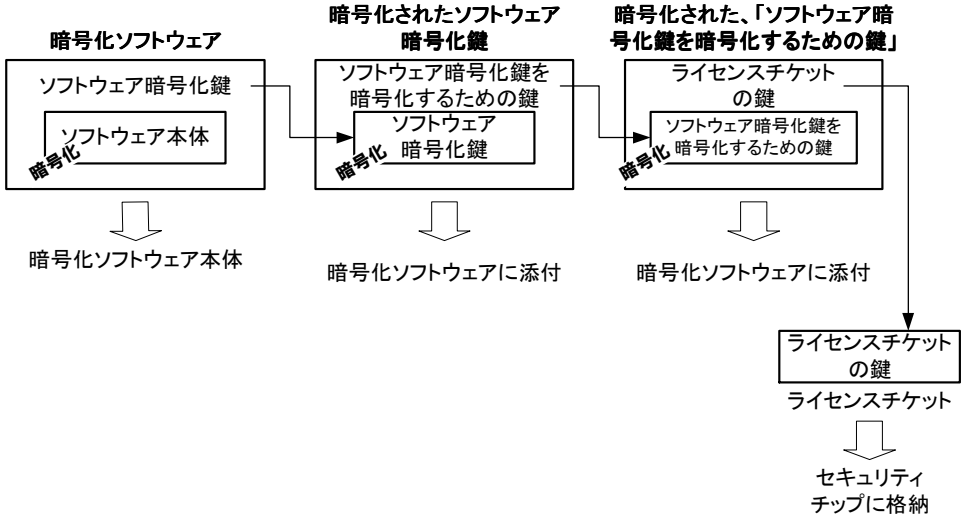


図 5-5-6 暗号化の依存関係

(2-2-8) 暗号化ソフトウェア結合機能

(a) 機能概要

ソフトウェア暗号鍵によるソフトウェア暗号化機能と標準コンテンツデータ暗号化要求機能にて生成した暗号化ソフトウェア本体、暗号化コンテンツデータを結合し、セキュアローダが実行可能な暗号ソフトウェアを生成する。

(b) 機能補足説明

暗号化ソフトウェア結合機能に関する補足説明を以下に示す。

(b-1) 生成する際に必要なデータ

- 暗号化ソフトウェア本体
  ソフトウェア暗号鍵により暗号化された暗号化ソフトウェア本体
- 暗号化コンテンツデータ
  標準コンテンツデータ暗号化機能で生成した暗号化標準コンテンツ本体

(b-2) 暗号化ソフトウェアヘッダーについて

上記の各データを連結した後、ヘッダー部分に暗号化ソフトウェアデータ、暗号化コンテンツデータ、ライセンスチケットに関する付加情報を格納する

(b-3) ライセンスチケットデータ

ライセンスチケット格納セキュリティチップ ID およびライセンスチケット ID

(b-4) 暗号化ソフトウェアの構成を下表に示す

表 5-5-5 暗号化ソフトウェア構成

項目	説明
暗号化ソフトウェアヘッダ	下記の各データの ID、オフセット値およびデータ長
暗号化ソフトウェアデータ	ソフトウェア暗号鍵によるソフトウェア暗号化機能で生成された暗号化ソフトウェア本体
暗号化コンテンツデータ	標準コンテンツデータ暗号化要求機能で生成された暗号化コンテンツ本体
ライセンスチケット付加データ	課金を行うライセンスチケットに関するデータ（課金および復号処理を行う際に、ライセンスチケットを指定するために必要なセキュリティチップおよびライセンスチケットの ID 等）

(2-3) セキュアローダによるライセンス制御機能

本機能は、暗号化ソフトウェアの実行時にライセンスチケットを利用して暗号化ソフトウェアを復号するとともに、課金処理を行うものである。

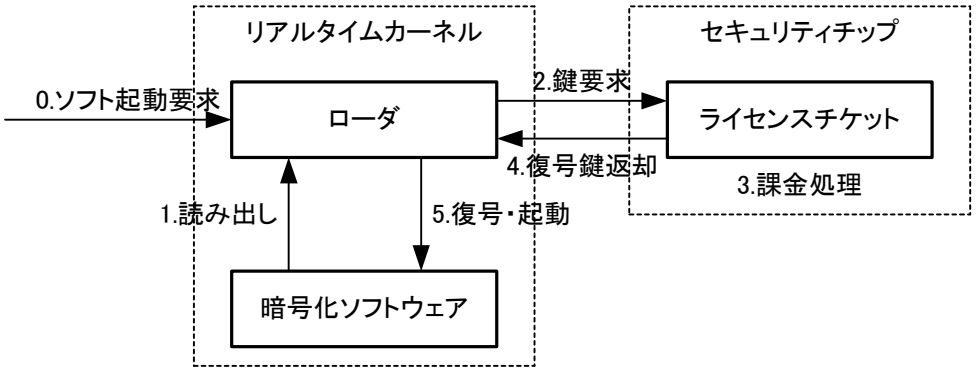


図 5-5-7 セキュアローダによるライセンス制御機能の概要

本機能は、さらに以下の各機能から構成される。

- 暗号化ソフトウェア解析機能
- 暗号化コンテンツ復号化要求機能
- ソフトウェア実行機能

これらの各機能について、詳細を以下に示す。

(2-3-1) 暗号化ソフトウェア解析機能

(a) 機能概要

ソフトウェア実行要求がなされると、ローダは暗号化ソフトウェアデータを読み込み、解析を行う。

(b) 機能補足説明

暗号化ソフトウェア解析機能に関する補足説明を以下に示す。

(b-1) 解析内容について

暗号化ソフトウェアのヘッダーを分析し、暗号化ソフトウェアを以下の 4 つの部分に分離できること

- 暗号化ソフトウェアヘッダ
- 暗号化ソフトウェアデータ
- 暗号化コンテンツデータ
- ライセンスチケット付加データ

(2-3-2) 暗号化コンテンツ復号化要求機能

(a) 機能概要

ローダは、セキュリティチップとトランザクションセッションを構築し、暗号化ソフトウェア解析機能を利用して取得した暗号化コンテンツデータをセキュリティチップ内のライセンスチケットを利用して復号する。復号処理の際に、セキュリティチップ上では課金処理が行われる。

(b) 機能補足説明

暗号化コンテンツ復号化要求機能に関する補足説明を以下に示す。

(b-1) ローダの要件について

(i) セキュリティチップに対して下記コマンドが発行可能であること

- eopn_tra() : トランザクションセッションの構築準備コマンド
  ローダ-セキュリティチップ間では暗号通信を行う必要があり、利用者が入力したパスワードを鍵として利用することも可能
- ecfm_tra() : トランザクションセッションの構築準備コマンド
- ecom_tra() : トランザクション終了コマンド
- eabo_tra() : トランザクション終了コマンド
- ecre_fil() : ファイルの作成コマンド
  「暗号化コンテンツデータ」をセキュリティチップ内に書き込む必要があるため
- edel_fil() : ファイルの削除コマンド
  任意のファイルの削除が行えること
- edec_fil() : ファイルのデータを復号するコマンド
  セキュリティチップに「標準コンテンツ形式のソフトウェア復号鍵」を書き込んだ際のファイル ID、および復号化したソフトウェア復号鍵が書き込まれる領域のファイル ID を指定し、暗号化された鍵の復号化の要求を行えること
- eupd_rec() : レコードのデータを更新するコマンド

(ii) 暗号化ソフトウェアのヘッダを読み取り、暗号化ソフトウェアの構成を把握できること

(iii) ローダは、ディレクトリ管理ファイルの解析が行えること

- セキュリティチップ内のディレクトリ管理ファイルを解析した結果、セキュリティチップ上に空きファイル領域が 2 箇所以上存在しない場合は、その旨を表示して本機能を終了し、トランザクションを abort すること(鍵実体を利用した復号を行うためには、空きファイル領域が 2 箇所以上必要なため)

(iv) ローダは復号化後コンテンツデータ形式のファイルの読み出し・解析が行えること

(v) ローダはトランザクションセッションのコミット、または、途中でエラーが

発生した場合にアボートが行えること

- アボートされた場合は、課金処理もロールバックされること

(b-2) ディレクトリ管理ファイルについて

ディレクトリ管理ファイルの構成を以下に示す。

- 各ファイルの File ID を利用して使用状況を管理する
- 使用状況は、「未使用」または、「使用中」フラグで管理する

File ID	使用状況	File ID	使用状況	File ID	使用状況	File ID	使用状況
…							
…							

図 5-5-8 ディレクトリ管理ファイル構成

(2-3-3) ソフトウェア実行機能

(a) 機能概要

ローダは、暗号化コンテンツ復号化要求機能により復号化されたソフトウェア暗号鍵を利用してソフトウェアを復号するとともに、復号化されたソフトウェアを実行する。

(b) 機能補足説明

(b-1) ソフトウェアの復号化について

暗号化コンテンツ復号化要求機能により復号化されたソフトウェア暗号鍵、および暗号アルゴリズムを利用して、ソフトウェア暗号鍵によりソフトウェアの復号が行えること

(b-2) ソフトウェアの実行について

復号化されたソフトウェアの main()関数を実行できること

5-6 超機能分散システム指向開発環境の研究開発

5-6-1 ユーザノードシステム

まず最初にユーザノードシステムが分散環境において重要なセキュリティ機能について述べる。

近年、映画・音楽といったアミューズメント用途に提供されるコンテンツに加え、個人がパーソナルコンピュータや小型情報機器で作成した情報をインターネットなどで配信したり、交換するという使い方が一般的になってきている。しかしそれとともに図 5-6-1 や図 5-6-2 に示すように配信元から得られた著作権

のあるコンテンツを無断で複製したり、その内容を改ざんしインターネット上で広く配布する事が社会問題にもなっている。

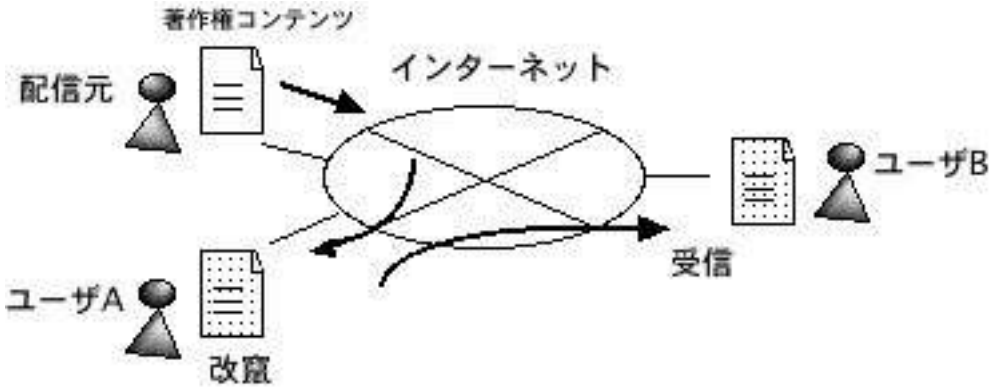


図 5-6-1 情報の改ざん

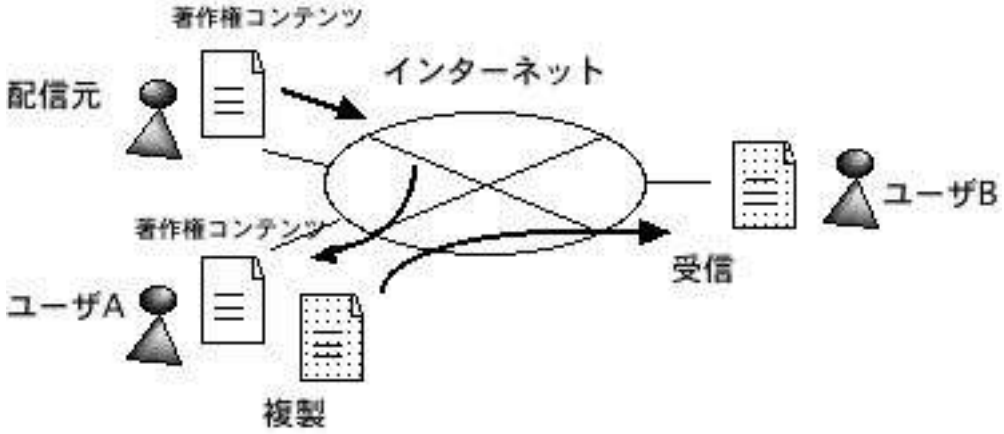


図 5-6-2 情報の複製

これは配信元にとっては課金の機会の喪失や、最終的に受信したユーザ B はそのコンテンツがどういった経緯で得られたものかをトレースすることはかなり困難であり本人の意思に関わらず違法な状態に陥る恐れがある。
このような背景から従来からのファイルフォーマットをなるべく変更することなく違法な複製を防ぐコンテンツ配信方法とデータの改ざんなどを防ぎセキュアなデータ交換方式の実現が早急に求められている。

そこで、ユビキタスコミュニケータにおいて U-Card と呼ばれる認証を導入しその解決をえることができた。たとえば、動画コンテンツ配信を例にとれば、配信者はインターネットなどを介して動画コンテンツを配信する。それをユーザは自分のユビキタスコミュニケータに受信する。そのコンテンツに対応した eTRON による認証を行って初めて再生を開始する。(図 5-6-3)

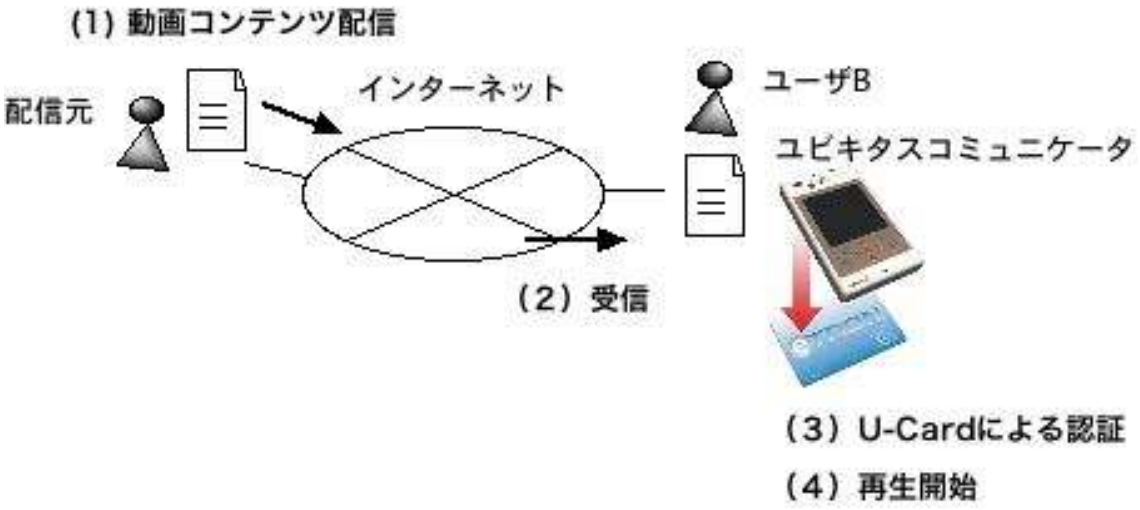


図 5-6-3 U-Card を用いた動画配信例

ここで U-Card とは耐タンパーチップをベースとして、物理的な実体の持つ一体性、 製造困難性、複製不能性、改竄困難性、携帯性などの性質を与えた特殊な性質を持ったデジタル情報であり、例えば図 5-6-4 のように不正な複製を行った場合は U-Card 内部に持つ情報が配信元から送付された物と一致しないため再生はできない。また内部が改竄された場合も同様に一致しないためそれを検知する事ができる。またいくら複製を行っても最後の U-Card による認証を得なければ再生することはできない。

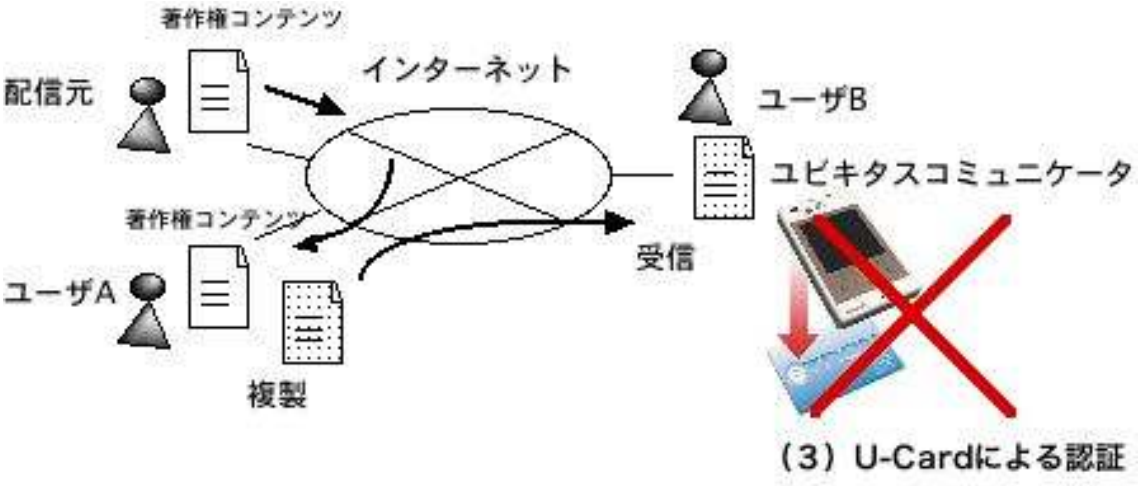


図 5-6-4 不正操作の検出

このように U-Card による認証機能を持つユビキタスコミュニケータはユビキタス環境において必須であるプライバシーに配慮したセキュアな情報環境に対応した数少ない情報機器ということができる。

またこの U-Card の技術をベースにさらに考え方を進め、ユビキタスコンピ

ューティングの世界の中でモノに対して識別をするためのユニークな識別子を設定するを策定する事とした。これは ucode と呼ばれる。ucode は自身の識別子を通知する手段を限定せず複数許し、例えば接触通信（IC カードなど）や非接触通信（RFID）バーコードなどを含んでいる。これらをそのカテゴリー分けしたものが表 5-6-1 である。

表 5-6-1 標準 ID タグカテゴリー分け

Class	名称	媒体
Class0	Visible ID	光学的 ID タグ（バーコード　２次元バーコード）
Class1	Low Level RFID	下位 RFID　読み出しのみできる RFID
Class2	High Level RFID	上位 RFID　読み書きのできる RFID
Class3	Low Level Smart Card	下位スマートカード　秘密鍵暗号が搭載されたスマートカード
Class4	High Level Smart Card	上位スマートカード　PKI 機能を用いた Smart Card
Class5	Low Level Active Chip	下位アクティブタグ　電池搭載の RFID スマートカード
Class6	High Level Active Chip	上位アクティブタグ
Class7	Security Box	セキュリティボックス
Class8	Security Server	セキュリティサーバ

上記カテゴリーにおいて研究所が昨年度までに開発した U-Card アーキテクチャは Class3 に相当する。また、世界的に次世代の物流や物流ソリューションの中で、非接触型で LSI の小型化が可能な高周波数帯を利用した RF タグが注目を浴びている。国内でも電波法で認可された 2.45GHz という高周波数帯域を利用する RF タグが登場しつつあり ucode を格納するタグもこの周波数帯を利用し上記表の Class1 のカテゴリに属する事となる。そのため、今後のユーザノードの開発環境としてすでに存在する U-Card（13.56MHz）のインタフェースに加えて 2.45GHz の RFID に対するインタフェースをもそなえ、マルチバンド対応にしたユビキタスコミュニケータを開発した。その概要を図 5-6-4 に示す。

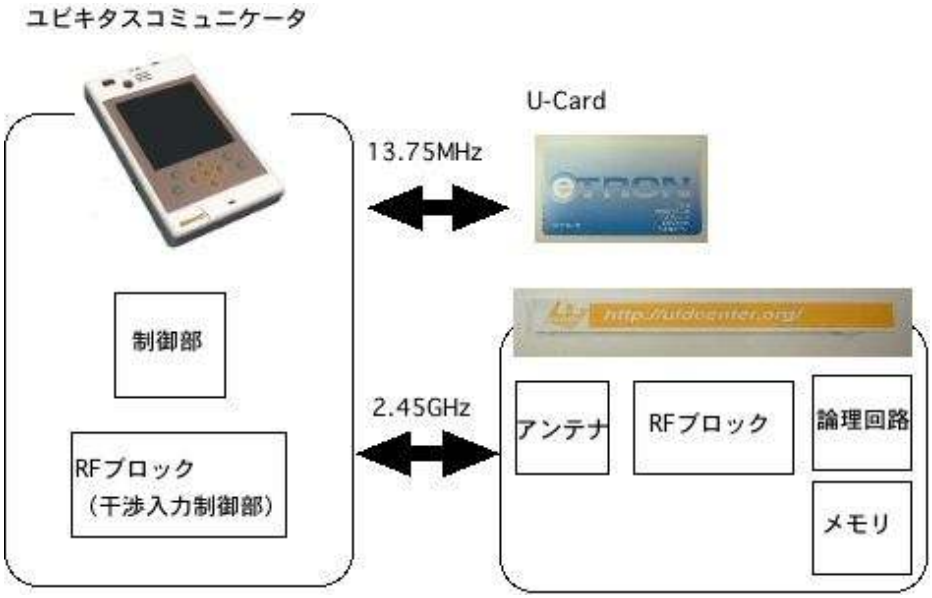


図 5-6-4 ユビキタスコミュニケータのマルチバンド対応

図 5-6-5　マルチバンド対応 ucode 読み取り装置の組み込み

実際に ucode を読み取りその内容を表示している所を図 5-6-6 に示す。

図 5-6-6　ucode 読み込みの例。

この例のように表示を行う事は図 5-6-7 の流れで行われる。

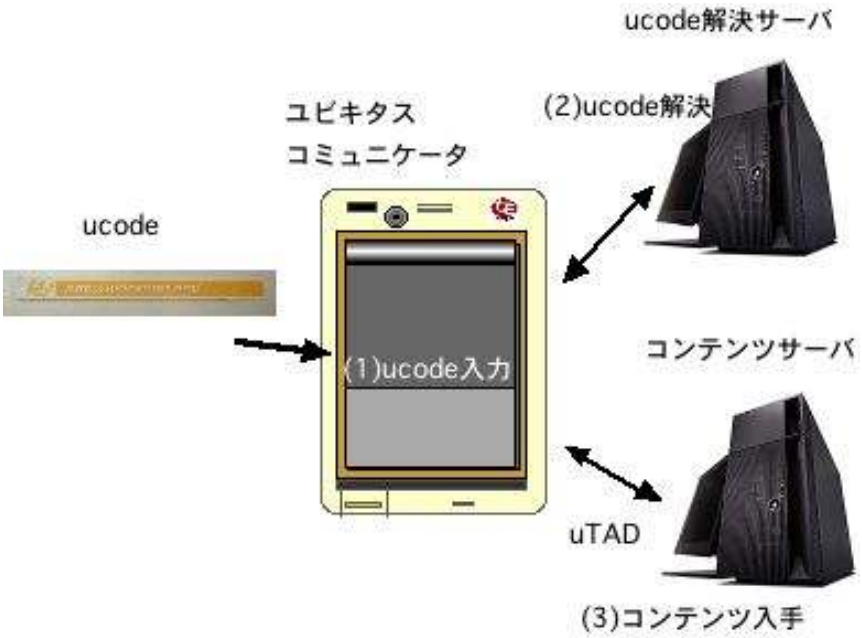


図 5-6-7 ucode フレームワーク

ucode にユビキタスコミュニケータをかざすと、そこで得られた ucode の情報を ucode 解決サーバと呼ばれる情報サービスサーバに問い合わせて ucode に対応付けられているコンテンツへのアドレスを得る。それを実際にモノの情報をもつコンテンツサーバに問い合わせる事によりモノの情報の実体を得る。

ここで得られるコンテンツは XML 技術を取り入れユビキタスコミュニケータで処理を考慮した uTAD と呼ばれるものである。その要素一覧を表 5-6- 2 に示す

表 5-6-2　uTAD 要素一覧

要素名称	内容
ubicontents	すべての要素のルート
contents	画面全体の属性を管理
image	画像を管理
sound	音声を管理
component	タップ制御などを管理
movie	動画を管理

開発者は、上記 uTAD の規格に従いモノの情報の定義を行う。それをコンテ

ンツサーバに登録した後、uCode を情報サービスサーバに ID を登録することにより、ucode を使った情報サービスを WEB サービスのようなコンテンツ提供する事ができるのである。

5-6-2 ミドルウェア

（ア） 目標

本研究の目的は、ユビキタス・ネットワーキングプロトコルに対して高抽象度のプログラミングインタフェースを提供するためのミドルウェアを開発することであり、詳細な目標は以下の通りである。

- ユビキタス・コンピューティング環境に適する、軽くソフトウェア規模の小さい Java Virtual Machine を開発する
- オブジェクト指向言語 Java のクラスライブラリとして開発する

上記目標の平成１５年度での達成状況について以下に示す。

（イ）成果概要

高抽象度のプログラミングインタフェースを提供するためのミドルウェアとして、U-Card のリアルタイムカーネル上で動作する Java Virtual Machine を開発した。

また、トロン文字列の制御および操作を行う API を定義した多漢字プロファイルの構築を行い、上記で構築した Java Virtual Machine への実装を行った。併せて、研究所にて開発されたユビキタスネットワーキングプロトコルライブラリ群を提供するために必要なクラスライブラリの設計・実装を行った。

（ウ） 開発した Java Virtual Machine ミドルウェアについて

(1) ミドルウェア概要

本ミドルウェアは、Java Virtual Machine をリアルタイムカーネル上で動作するように移植したものである。Java Virtual Machine は下記の Configuration および Profile から構成される。

- CDC:Connected Device Configuration
- FP:Foundation Profile

また、この CDC/FP は、以下のような特徴を持つ。

- Java 2 Standard Edition ベースのクラスライブラリ
- より上位のプロファイルとは異なり、GUI サポートはない
- CLDC（Connected Limited Device Configuration）1.0 と互換性のあるライブラリ

(2) ミドルウェア前提条件

本ミドルウェアは、以下の環境を前提とする。

開発ホストマシン環境

- シリアルポートを持つ PC/AT 互換機
- gmake 等を含む標準的な開発環境を備えた Linux
- U-Card 専用開発環境
- Java Development Kit 1.3.1

開発ターゲットマシン環境

- U-Card
- リアルタイムカーネル

(3) 動作検証結果

下記プログラムを実装し、動作検証を実施した。

- 文字列操作プログラム（GUI）
- 文字列変換プログラム
- ネットワークを利用したプログラム

以下に、各プログラムの動作検証結果を記す。

(3-1) 文字列操作プログラム（GUI）

(3-1-1) プログラム

リアルタイムカーネルの GUI スクリプトを利用したデモプログラム。GUI スクリプトを通じて Java プログラムが簡単な文字列操作を行うもの。（実際には、GUI スクリプトと Java プログラムのメッセージ通信を行っている。）

(3-1-2) 実行結果

「操作開始」ボタンをクリックすると、指定された操作を文字列に対して行い、その結果を表示する。

可能な操作は下記の通りである。

- 文字列の追加

・　文字列の削除
・　文字列の挿入
・　文字列の置換
・　文字列の逆順変換
・　文字列から部分文字列を抽出

図 5-6-8 実行結果

上記写真は、文字列１ ”333” に文字列２ ”111” を追加した結果として ”333111” が出力されたところ。

(3-2) 文字列変換プログラム

標準入力から入力された文字列を、小文字を大文字にした上で出力するプログラム

(3-2-1) プログラム

```
import java.io.*;


public class UpperCaseTest {
  public static void main(String[] args) throws IOException {
```

```
    char c;
    String str = new String();
    while ((c = (char)System.in.read()) != '¥n') {
      str = str + c;
    }
    System.out.println(str.toUpperCase());
  }
}
```

(3-2-2) 実行結果

“This is a test!”と入力された文字列が” !HIS IS A TEST!!”に変換されている。ここで表示されている「!」は、トロンコードの言語指定コードによるものである。このように、トロンコードの言語指定コードの関係で、文字列の先端と末端は正しく表示されない。

```
[/SYS/WORK/java/test]%                                  /SYS/WORK/java/bin/cvm
-Djava.class.path=/SYS/WORK/java/test UpperCaseTest
GC[SS]: Initialized semi-space gen for generational GC
Size of *each* semispace in bytes=1048576
Limits of generation = [0x4fe200,0x6fe200)
First semispace = [0x4fe200,0x5fe200)
Second semispace = [0x5fe200,0x6fe200)
GC[MC]: Initialized mark-compact gen for generational GC
Size of the space in bytes=3145728
Limits of generation = [0x6fe200,0x9fe200)
GC[generational]: Auxiliary data structures
heapBaseMemoryArea=[0x4fe008,0x9fe208)
cardTable=[0x9ff008,0xa01808)
objectHeaderTable=[0xa02008,0xa04808)
summaryTable=[0xa05008,0xa0f008)
!ecurity properties not found. using defaults.!
This is a test!
!HIS IS A TEST!!
```

(3-3) ネットワークを利用したプログラム

サーバープログラムは、クライアントから接続を受けつけ、送られて来た文字列を大文字に変換して送り返す。なお、ポートはここでは 12345 を用いている。

(3-3-1) プログラム

(a) サーバサイドプログラム

```
public class ServerImpl {
   public static void main(String[] args) throws IOException {
      ServerSocket ss = new ServerSocket(12345);
      for(;;) {
         Socket soc = ss.accept();
         InputStream in = soc.getInputStream();
         OutputStream out = soc.getOutputStream();
         byte[] buf = new byte[256];
         in.read(buf);
         System.out.println(soc.getInetAddress() + " " + new String(buf));
         out.write((new String(buf)).toUpperCase().getBytes());
         soc.close();
      }
   }
}
```

(b) クライアントサイドプログラム

クライアントプログラムは、サーバーに接続し標準入力より入力された文字列を送信する。その後サーバから送られて来た文字列を標準出力に出力する。なお、ここではサーバのアドレスおよびポート番号を指定する必要がある。

```
import java.io.*;
import java.net.*;
public class ClientImpl {
   public static void main(String[] args) throws IOException {
      Socket soc = new Socket(InetAddress.getByName("?.?.?.?"),12345);
```

```
        InputStream in = soc.getInputStream();
        OutputStream out = soc.getOutputStream();
        String str = new String();
        byte[] buf = new byte[256];
        int c;
        while((c = System.in.read()) != '¥n') {
          str = str + (char)c;
        }
        out.write(str.getBytes());
        str = new String();
        while((c = in.read()) != -1) {
          str += (char)c
        };
        System.out.println(str);
    }
}
```

ここでは、例として、

- 開発ホストとリアルタイムカーネルを搭載した U-Card が同一ネットワーク上にある
- 開発ホスト上でサーバプログラムを実行する
- リアルタイムカーネル上でクライアントプログラムを実行する

ということを前提としている。この場合は、クライアントプログラム中のサーバアドレスには、開発ホストのアドレスを指定する。

(3-3-2) 実行結果

実行すると以下のように表示される。文字列変換プログラムと同様、トロンコードの言語指定コードの関係で、文字列の先端と末端は正しく表示されない。

```
[/SYS/WORK/java/test]% /SYS/WORK/java/bin/cvm -Djava.class.path=/SYS/WORK/java/test
ClientImpl
GC[SS]: Initialized semi-space gen for generational GC
Size of *each* semispace in bytes=1048576
Limits of generation = [0x4fe200,0x6fe200)
```

```
First semispace = [0x4fe200,0x5fe200)
Second semispace = [0x5fe200,0x6fe200)
GC[MC]: Initialized mark-compact gen for generational GC
Size of the space in bytes=3145728
Limits of generation = [0x6fe200,0x9fe200)
GC[generational]: Auxiliary data structures
heapBaseMemoryArea=[0x4fe008,0x9fe208)
cardTable=[0x9ff008,0xa01808)
objectHeaderTable=[0xa02008,0xa04808)
summaryTable=[0xa05008,0xa0f008)
!ecurity properties not found. using defaults.!
This is a sample!
!?THIS IS A SAMPLE!!
[/SYS/WORK/java/test]%
```

(4) 本ミドルウェアの制約事項

本ミドルウェアの制約事項を以下に示す。

- ダイナミックリンク機能はメインプログラム(すなわち本 Java Virtual Machine)のみサポートする。
- 全てのファイルパスは絶対パスで記述する必要がある。
- JNI ネイティブ・メソッドの呼び出し部分の引数に制限がある。
- ターミナル上で Virtual Machine を実行すると、標準(エラー)出力に、文字「！」が表示される。
  Java のコンバータが外部に出力を行う際に、トロンコードから EUC にコード変換を行うとき、言語指定コードをそのまま出力しているためである。

（エ） 開発したクラスライブラリ群について

(1) 多漢字プロファイル

トロン文字列の制御および操作を行う API を定義した多漢字プロファイルを設計し、上記で開発した Java Virtual Machine 上に実装した。

多漢字プロファイルは以下のクラス群より構成される。

トロンコードの基本的な文字列操作（org.tron.lang パッケージ）

- TChar クラス
  public final class TChar extends java.lang.Object
  トロンコードを扱うためのクラス
- TString クラス
  public final class TString extends java.lang.Object
  トロンコード文字列を扱うクラス
- TStringBuffer クラス
  public final class TStringBuffer extends java.lang.Object
  トロンコード文字列バッファを扱うためのクラス

トロンコードの I/O 操作（org.tron.io パッケージ）

- TFileReader クラス
  public class TFileReader extends TInputStreamReader
  トロンコード用 FileReader
- TFileWrite クラス
  public class TFileWriter extends TOutputStreamWriter
  トロンコード用 FileWriter
- TInputStreamReader クラス
  public class TInputStreamReader extends TReader
  トロンコード用 InputStreamReader。内部コードにトロンコードを用いる以外は java.io.InputStreamReader と 同じ動作を行う
- TOutputStreamWriter
  public class TOutputStreamWriter extends TWriter
  トロンコード用 OutputStreamWriter
- TReader クラス
  public abstract class TReader extends java.lang.Object
  入力ストリームの抽象クラス。内部コードがトロンコードである以外は java.io.Reader と同等
- TWriter クラス
  public abstract class TWriter extends java.lang.Object
  トロンコード用 Writer。

多漢字プロファイルは Java のコアパッケージとの互換性を持たせるため、外部文字コードとトロンコードとの変換を行う機能を持つとともに、トロンコードをバイナリデータとして扱う以外に他のアプリケーションとの連携を図るためテキスト形式のトロンコードとしても扱う。今後は、これらのライブラリ上にさらに多くの高抽象度、高付加価値のライブラリ提供を行っていく予定である。

(2) ユビキタス・ネットワーキングプロトコルライブラリ群

ユビキタスネットワーキングプロトコルを提供するために必要なクラスライブラリの設計・実装を行った。ユビキタス・ネットワーキングプロトコルライブラリ群は、下記のライブラリから構成される。

- 認証ライブラリ
- 暗号ライブラリ
- CRL ライブラリ
- アドレス管理ライブラリ
- コマンドライブラリ
- 上位の高抽象度ライブラリ
  - 電子チケット発行ライブラリ
  - 電子図書券発行ライブラリ
  - 電子バリュー発行ライブラリ

以下に、各ライブラリの詳細について記す。

(2-1) 認証ライブラリ

認証に関する処理を行うライブラリである。具体的には、以下の機能を提供する。

- ユビキタス・ネットワーキングプロトコル仕様に基づく公開鍵ペア生成
- 自己署名公開鍵証明書（認証サーバ用公開鍵証明書）生成
- 認証サーバ以外のユビキタス・ネットワーキングプロトコルを利用する各ノード用の公開鍵証明書生成
- ユビキタス・ネットワーキングプロトコル仕様に基づく公開鍵証明書の発行
- 署名の検証

(2-2) 暗号ライブラリ

ユビキタス・ネットワーキングプロトコルで通信する際に必要となる暗号ロジ

ックを提供するライブラリである。以下の機能を有する。

(a) 共通鍵暗号ライブラリ

- DES
- 3DES
- Rijndael
- Hierocrypt-3
- CAMELLIA

(b) MAC 計算

- CRC-CCITT
- HMAC with MD5

(2-3) CRL ライブラリ

破棄された証明書リストを管理する際に必要となる処理を提供するライブラリ群。以下の機能を提供する。

- CRL 登録・削除
- CRL チェック実行

(2-4) アドレス管理ライブラリ

eTRON ID とネットワーク層の経路制御アドレス（IP アドレス）の組を管理するライブラリ。以下の機能を提供する。

- IP アドレスと eTRON ID の組の登録
- 上記登録の削除
- eTRON ID による IP アドレスの検索

(2-5) コマンドライブラリ

「基盤通信システムの研究開発」により開発されたユビキタス・ネットワーキングプロトコルのアトミックな各命令のライブラリ。以下の機能を提供する。

- セッション通信
- トランザクション通信
- ファイル操作
- レコード操作
- ポーリング操作
- ファイルモード操作

(2-6) 上位の高抽象度ライブラリ

高抽象度の Java ライブラリとして、電子チケット発行ライブラリ、電子図書

券発行ライブラリ、および、電子バリュー発行ライブラリの開発を行った。いずれのライブラリも、ユビキタス・ネットワーキングプロトコル群のライブラリを通して、様々なサービスを提供する高抽象度のライブラリである。このライブラリでは、本研究所で研究開発されたセキュア IC チップに対してセキュアに電子チケット、電子図書券、および、電子的価値情報を発行・流通できることができる高抽象度のライブラリである。

本ライブラリは、電子的価値情報をセキュアに流通させる高抽象度のアプリケーションとして、インターネット上に存在するチケット販売サイト（価値情報プロバイダ）から購入したチケットを、ユビキタス・ネットワーキングプロトコルを利用して、指定するセキュア IC チップへ直接または間接的にダウンロードさせる際に利用される。

（オ） まとめ

Java Virtual Machine、およびクラスライブラリを開発したことにより、移植性やネットワークの接続性に富んだ分散指向言語である Java がリアルタイムカーネル上で実行可能となり、P2P ミドルウェア等の既存のソフトウェアが利用できたり、あるいは新規の応用開発が容易となった。平成 16 年度以降も Java を利用した応用開発を行っていく予定である。

5-7 総括

平成 15 年度は、本委託研究の 3 年目にあたり、前述したように多岐に渡る研究活動を行った。以下にその成果をまとめる。

5-7-1 【サブテーマ 1】セキュアコンピューティングの基盤となるハードウェア

(1) 高性能なハイエンド型のセキュアチップ

ハイエンド型セキュアチップの改良バージョンとして、16 ビット CPU を用いたセキュアデータキャリアチップを開発し、より安全なユビキタス通信基盤を実現することに成功した。本チップとそれを利用するためのライブラリ群により、次年度以降効率的なユビキタス応用が実現可能である。

(2) 超小型セキュアチップ

RFID の貼り付け環境に応じた特性の地道な知見を得ることができ、今後様々な応用に当該技術を適用するときに十分活用できる。地味な知見であったが、実用にむけては極めて重要なものである。

5-7-2【サブテーマ２】基盤通信システム

(1) シームレス通信の研究

シームレス通信を実現する方式を考案した。これによりユーザノードの移動時にも可能な限り通信を提供し、また複数の通信手段が利用可能な場合により適した通信手段により通信することが可能となり、今後のユーザノード上でのユビキタス応用の実現が容易となる。本考案方式は平成 16 年度に実証を行う予定である。

(2) ユビキタス情報提供・制御用プロトコルの研究

ユビキタス情報提供・制御用プロトコルとそれを実行するハードウェアが完成し、リアルタイム性やセキュリティが強化されたユビキタス通信が実現できる。

(3) 実世界データ研究・Everything ID 研究

番号体系と番号解析プロトコルを開発し、RFID 等を利用したユビキタス応用が実証可能となった。今後各種実証実験等で適用していく予定である。

5-7-3 【サブテーマ３】ユーザノードシステム

(1) ユーザノードシステムの開発

移動型や固定型など多岐にわたるユーザノードを開発し、それらを利用して、各種のユビキタス応用の実現が可能となった。平成１６年度以降これらのユーザノードで動作する応用を開発していく予定である。

5-7-4 【サブテーマ４】サーバノードシステム

開発した 16 ビット版のセキュアプロトコルを利用して、より高信頼なセキュア通信基盤が確立できた。また、アドレス解決サーバやセキュア発行サーバ等のサーバ群により、電子的な価値の流通基盤が完成したことになる。これとユビキタス ID サーバを組み合わせることで、RFID やバーコード等を利用したユビキタスオブジェクトの管理システムが実現できる。

5-7-5【サブテーマ５】システム統合技術

(1) 次世代通信プロトコルと既存ネットワークプロトコルとの相互接続技術

コンテンツ配布流通用ソフトウェアを開発し、コンテンツ流通の実証実験等を行い評価を進めていく予定である。

5-7-6【サブテーマ６】超機能分散システム指向開発環境の整備

(1) ユーザノードシステム

RFIDインタフェースを有するコンパクトな開発環境としてUCの開発が完了したことによって、今後ユビキタスネットワーキング研究の代表的なユーザノードプラットフォームが構築できたことになる。これによって、次年度以降の、応用研究や通信プロトコルの、ユーザインタフェースの研究などに活用できると考えている。

（２）ミドルウェア

移植性やネットワークの接続性に富んだ分散指向言語である Java が実行可能となり P2P ミドルウェア等の既存のソフトウェアが利用できたり、あるいは新規の応用の開発が容易となった。平成 16 年度以降も Java を利用した応用開発を行っていく予定である。

添付資料１　研究発表、講演、文献等一覧　（平成 15 年度）

学術論文

[1]　下川功, 宮崎祐行, 志田雅昭, 大熊康介, 早川幹, 越塚登, 坂村健 :「センサーネット向け無線通信システムにおけるマルチプルアクセス方式の検討」, 電子情報通信学会2004年総合大会, 2004年3月.

[2]　李海量, 越塚登, 坂村健 :「コンテキスト情報を利用して曖昧な音声入力の意味解決をする音声ユーザインタフェースシステム」, 第66回情報処理学会全国大会, 2004年3月.

[3]　松沢敬一, 新堂克徳, 越塚登, 坂村健 :「管理者による視認型認証を支援するICカードを用いた本人確認システム」, 第66回情報処理学会全国大会, 2004年3月.

[4]　別所正博, 鵜坂智則, 越塚登, 坂村健 :「ユビキタス環境における緊急避難経路提示システムの提案」, 第66回情報処理学会全国大会, 2004年3月.

[5]　佐藤, 豊山, 田中, 越塚登, 坂村健:「組込みシステムのプラットフォームの標準化によるソフトウェア資産の再利用性向上の評価」, 第66回情報処理学会全国大会, 2004年3月.

[6]　渡辺伸吾, 西山智, 服部元, 小野智弘, 越塚登, 坂村健:「既存ルータ混在環境におけるモバイルIPハンドオーバの高速・高信頼化の提案」, 第66回情報処理学会全国大会, 2004年3月.

[7]　宮崎真悟, 石川千秋, 鵜坂智則, 小俣三郎, 越塚登, 坂村健:「組込み機器に秘密共有機能を提供するSIMカード型セキュアチップの開発」, 第66回情報処理学会全国大会, 2004年3月.

[8]　西山智, 渡辺伸吾, 服部元, 小野智弘, 越塚登, 坂村健:「モバイル端末における応用の要求に応じた通信メディアの使い分け方式の提案」, 第66回情報処理学会全国大会, 2004年3月.

[9]　Ken Sakamura: “T-Engine－The Open Development Platform fro Ubiquitous Computing”, サイバーアシストコンソーシアム第２回国際シンポジウム, pp. 1～15, 2003.

[10]　坂村健,「ディジタルミュージアムからユビキタスミュージアムへ」, 人工知能学会誌5月号, Vol. 18, No. 3, pp. 259～266, 2003年.

[11]　坂村健:「解説：特集『電脳都市』２」, 計測と制御１, 第43巻, 2004年, pp. 52～58.

[12]　Ken Sakamura:“Ucode Architecture and RFID”, in Proc. 2004

RFID International Symposium (Korea), 2004年, pp. 3〜24.

[13] 坂村健: 「ユビキタス・コンピューティング社会にむけて」, シリコンシーベルトサミット2004福岡, 2004年, pp. 3〜10.

[14] Ken Sakamura: “Ubiquitous Computing: Making It a Reality”, ITU TELECOM World2003, Geneva Palexpo, Geneva, Oct. 13, 2003, pp. 1〜9.

[15] 坂村健 :「ユビキタスコンピューテング環境の実現にむけて」, Microwave Workshop and Exhibition (MWE 2003), 2003年.

[16] Shingo Watanabe, Satoshi Nishiyama, Noboru Koshizuka, Ken Sakamura: “Location Detection Method for Everyday Objects Using Contactless IC Cards”, Microwave Workshop and Exhibition (MWE 2003), Nov. 2003, pp. 245〜250.

[17] Katsunori SHINDO, Noboru Koshizuka, and Ken Sakamura: “Ubiquitous Digital Museum Using Contactless Smart Cards”, Microwave Workshop and Exhibition (MWE 2003), Nov. 2003, pp. 251〜256.

[18] Ken Sakamura and Noboru Koshizuka: “ Technologies for Computing Everywhere Environments ” , Korea Information Processing Society Review, July, 2003, pp. 11〜22.

[19] 渡辺伸吾, 西山智, 服部元, 小野智弘, 越塚登, 坂村健: 「ユビキタス環境のための非接触ICカードを使用した位置検出方式の実装と評価」, FITS2003 第二回情報科学技術フォーラム研究報告, 2003年9月.

[20] 西山智, 渡辺伸吾, 服部元, 小野智弘, 越塚登, 坂村健: 「屋内用センサネットワーク用ネットワークプロトコルの実装」, FITS2003 第二回情報科学技術フォーラム研究報告, 2003年9月.

[21] Noboru Koshizuka and Ken Sakamura: “T-Engine Project: The Open Platform Project for Ubiquitous Computing”, in Proc. First International Conference on Ubiquitous Computing (ICUC 2003), pp. 185-190, 2003.

[22] Ken Sakamura and Noboru Koshizuka: “T-Engine: The Open, Real-time Embedded-Systems Platform for Ubiquitous Computing”, in Proceedings of the 2003 Symposium on VLSI Circuits, June 2003.

[23] 渡辺伸吾, 西山智, 服部元, 小野智弘, 越塚登, 坂村健: 「ユビキタス環境のための非接触ICカードを使用した位置検出方式」, 第一回ユビキタスコンピューティングシステム研究会, 情報処理学会, 2003.

解説・著書等

[1] 坂村健: 「環境とデザイン-05 情報環境から建築を考える」, 新建築78, pp. 052～055, 2003.
[2] 坂村健: 「組み込みエンジニアへおくる最新情報、ユビキタス・コンピューティングにおけるT-Engineの動向」, EPO (Electronics Product Digest), pp. 4～5, 2003.
[3] 坂村健: 「総特集　ITで医療が変わる：ユビキタス・コンピューティング実現のために」, 月刊 新医療, No.344, pp. 52～56, 2003.
[4] 坂村健:「特集　ユビキタス・コンピューティングと物流システム」, ロジスティクスシステム, 12 (6), pp. 6～9, 2003.
[5] 坂村健: 「正論　開かれた土俵こそが相互繁栄の道　歴史的なMSとトロンの提携劇」, 産経新聞（10月3日）, p 13, 2003.
[6] 越塚登:「ユビキタスIDセンターとその活動」,「ユビキタス社会のRFID徹底解説」, 電子ジャーナル, 2003年.
[7] 坂村健,「日本発ユビキタスの目標は、監視社会ではなく究極の便利社会である」, 日本の論点2004, pp. 412～417, 2003.
[8] 越塚登 :「ユビキタスコンピューティングとトロン」, 情報通信i-Net, 第7号, 数研出版, 2003年6月.
[9] 坂村健,:「大事に育てたい未来技術ユビキタス、社会的合意と環境整備を急げ、産経新聞（5月24日), p. 12, 2003.
[10] 坂村健,越塚登:「ユビキタスIDセンターの取り組み」, 月刊バーコード, vol. 16, no. 5, 日本工業出版, 2003年4月, pp. 15～20.

※　その他多数

講演

[1] 坂村健, 越塚登:「ユビキタスコンピューティングの世界で、何ができるのか？」ユビキタスIDセミナー, ユビキタスIDセンター・日経ＢＰ社, 2003年4月.
[2] 坂村健：OA学会, 中央学院大学, 2003年4月.
[3] 坂村健：RSA Conference2003 Japan, 東京国際フォーラム, 2003年6月.
[4] 坂村健：ACM 日本支部総会および特別講演会, 東京理科大学森戸記念館, 2003年7月.
[5] 坂村健：TRONプロジェクトの20年, 安田講堂, 2003年7月.
[6] 坂村健：建築学会(1400-1450), 中部大学, 2003年9月.

[7] 坂村健：Asian Enterprise Open Source Conference, 2003, Singapore, October, 2003.
[8] 坂村健：データベース研究会，日本科学未来館, 2003年11月.
[9] 坂村健：第一回アジアユビキタス会議基調講演，パークタワーホール, 2003年12月.
[10] 坂村健 : TRONSHOW2004基調講演，東京国際フォーラム, 2003年12月.
[11] 坂村健：第2回武田シンポジウム，東大武田ホール（6F), 2004年2月.
[12] 越塚登 :「ユビキタスID技術」，電子情報通信学会QoSワークショップチュートリアル, 2004年2月.
[13] 越塚登 :「T-Engineとユビキタス ID」，日本学術会議シリコン超集積化システム第165委員会, 2004年1月.
[14] 越塚登 :「T-Engineとユビキタス ID」，第6回 CEST 技術セミナー「TRONプロジェクトの最前線: T-Engineとユビキタス ID」，組込みシステム開発技術研究会（CEST），豊橋商工会議所, 2003年10月
[15] 越塚登: 「ユビキタスIDの最新動向」，日経BP・RFIDユーザーフォーラムSpring 2004 “無線ＩＣタグ実用化の幕開け”, 2004年3月.
[16] 越塚登: 「ユビキタスID技術を用いた青果物トレーサビリティシステムの構築」，食品トレーサビリティシステム普及推進セミナー, 2004年3月.
[17] 越塚登 :「T-EngineとユビキタスIDプロジェクト-ユビキタス社会を目指して」，オープンソリューションパートナーズグループ（OSPG）講演会, 2004年2月.
[18] 越塚登 :「ユビキタスIDセンター―次世代の情報技術基盤の確立に向けて―」, ECOM, 2004年1月.
[19] 越塚，他: 「組込み型リアルタイムOS入門」，トロン協会，にいがた産業創造機構，ハイブ長岡（新潟県長岡市）, 2003年12月.
[20] 越塚登:「ユビキタスID技術－次世代の情報技術基盤の確立に向けて－」, 社団法人日本自動認識システム協会RFID部会, 2003年12月.
[21] 越塚，他 :「使ってみようT-Engine」, TRONSHOW 2004, 2003年12月.
[22] 越塚，他 :「TRONポータビリティーWG」, TRONSHOW 2004, 2003年12月.
[23] 越塚登 :「T-Engineフォーラムの活動」, TRONSHOW 2004, 2003年12月.
[24] 越塚登 :「T-Kernelのオープン化」, TRONSHOW 2004, 2003年12月.
[25] 越塚登 :「ユビキタスコミュニケーター」, TRONSHOW 2004, 2003年12月.
[26] 越塚登:「ユビキタスID技術－次世代の情報技術基盤の確立に向けて－」,

（社）自動車技術会中部支部・技術講演会「21世紀を担う情報・通信技術と自動車」，名古屋, 2003年11月.

[27] 越塚登 :「ユビキタスＩＤ技術とその応用事例」，まちと人のセキュリティーシンポジウム2003,（社）日本能率協会，東京ビッグサイト，2003年11月.

[28] 越塚登，豊山祐一 :「T-Engineとユビキタス ID技術の最新動向－T-Engine Project－」, Embedded Technology 2002 組込み総合技術展特別講演, 2003年11月.

[29] 越塚，他 :「T-Engineセミナー」, Embedded Technology 2002 組込み総合技術展チュートリアル, 2003年11月.

[30] 越塚登 :「ユビキタスコンピューティングの実現にむけて」，日本フォーム印刷工業連合会・技術セミナー, 2003年11月.

[31] 越塚登 :「ユビキタスＩＤセンターの活動～ユビキタス環境の実現にむけて～」，第55回テレコム技術情報セミナー，財団法人テレコム先端技術研究支援センター, 2003年10月.

[32] 越塚登 :「ユビキタスコンピューティングと物流システム―ユビキタスID技術が創る未来―」，ロジスティックスフォーラム関西2003，大阪, 2003年10月.

[33] 坂村健，越塚登，西山智:「ユビキタスコンピューティング環境を実現する基盤ネットワークプロトコルの研究開発」，平成15年度通信・放送機構（TAO）研究発表会.

[34] 越塚登:「トロンが実現するどこでもコンピュータの世界」，滋賀県高度情報化推進会議, 2003年7月.

[35] 越塚登，他:「ユビキタスコンピューティングの基礎技術『T-Engine』と『ユビキタスID』の現状と展望」，第6回組込みシステム 開発技術展 (ESEC: Embedded Systems Expo. & Conference in Tokyo 2003), 2003年7月，東京ビッグサイト.

[36] 越塚登:「ユビキタスネットワーキングとユビキタスコンピューティング」, JEITA，ユビキタスネットワーク部会, 2003年7月.

[37] 越塚登:「ユビキタスコンピューティングとユビキタスIDセンター」, JEITA総会, 2003年5月.

[38] 越塚登:「ユビキタスID：技術的内容と方向性」，日経コンピューター・セミナー，「ICタグ」の全貌、最新技術動向から応用まで，日経ＢＰ社, 2003年5月.

※　その他多数