

新アクセスシステムアーキテクチャ (FASA)
ホワイトペーパー

Ver. 1.0

NTT アクセスサービスシステム研究所

2016/05/31

目次

要旨

1. はじめに
 - 1.1. 本資料の目的
 - 1.2. FASA 概要
 - 1.3. 本資料における FASA アプリケーション API の規定範囲
 - 1.4. 用語と定義及び略語
 - 1.5. 参考文献
2. 参照アーキテクチャ
 - 2.1. 論理モデル
 - 2.2. システム構成例
3. ユースケース
 - 3.1. DBA の入替によるマルチサービス収容の実現
 - 3.2. 省電力機能の入替による通信事業者独自仕様の実現
 - 3.3. プロテクション機能の入替による通信事業者独自仕様の実現
4. アクセスシステムの主要機能と FASA アプリケーション化の対象
5. API 規定
 - 5.1. PON 主信号処理機能
 - 5.2. PON アクセス制御機能
 - 5.2.1 DBA
 - 5.3. L2 主信号処理機能
 - 5.4. 保守運用機能
 - 5.5. PON マルチキャスト機能
 - 5.6. 省電力制御機能
 - 5.7. 周波数/時刻同期機能
 - 5.8. プロテクション機能

要旨

通信の用途や利用形態の多様化に伴い、通信事業者が通信サービスをユーザに直接提供するB2C型通信サービスに加えて、様々なサービス事業者を介して通信サービスを提供するB2B2C型通信サービスの比重が、大きくなってきている。NTTにおいても、2014年に「光コラボレーションモデル」¹を発表し、様々なビジネスプレイヤーとの共創により新たなサービスを提供していくこととしている。このような状況の変化に伴い、B2B2Cの「ミドルB」に相当するサービス事業者やエンドユーザからの多様な要求(帯域、遅延、コスト、信頼性やそれらの粒度等)に対して、アクセスシステムにおいても柔軟かつ迅速に対応可能とする必要がある。

従来はこのような個々の要望に対して柔軟に対応するというサービス形態ではなかったこともあり、アクセスシステムはサービス毎の専用装置として開発が行われていた。このため従来の装置では、今後ますます重要となると思われる、サービス毎に種類の異なる要求への対応や、通信サービスのビジネスモデルの変化により、多様化する要求に対応するための機能の追加や入替には、装置全体の再開発が必要であるなど、迅速な対応が困難な状況であった。

このような多様な要求に対応し、柔軟かつ迅速なサービス提供を実現するために、NTTではNetroSphere構想²を提唱しており、ネットワークを構成する機能を部品化し、必要に応じて自由自在に組み合わせて構成する技術の研究開発に取り組んでいる。アクセスネットワークに関しても、同様のコンセプトに基づく新アクセスシステムアーキテクチャFASA (Flexible Access System Architecture)³を発表したところである。

FASAではNetroSphere構想の考え方にに基づき、アクセスネットワーク装置の有する各機能を部品化し、それらの柔軟な組み合わせを可能とする構成を提案している。サービス毎あるいは通信事業者毎に異なる機能は、入替可能なソフトウェア部品(FASAアプリケーション)で実現し、汎用化した入出力インタフェースを提供する基盤(FASA基盤)上でそれらのソフトウェア部品を動作させる。このような構成により、柔軟な機能の追加や入替を実施可能とし、多様な要求に迅速かつ柔軟に対応するアクセスネットワークを実現する。

図1に、FASAによる部品化の構成例を示す。

構成1は、サービス毎あるいは通信事業者毎に異なる要求に応えるために入替が必要な機能(FASAアプリケーション)をソフトウェア部品化した構成である。

構成2は、FASAアプリケーション以外(FASA基盤)の機能もソフトウェア部品化の対象とすることで、より汎用的なハードウェアを用いてアクセスネットワーク装置を実現する構成である。

¹ <http://www.ntt.co.jp/irdoc/kaijishiryu/pdf/2014/008.pdf>

² <http://www.ntt.co.jp/news2015/1502/150219a.html>

³ <http://www.ntt.co.jp/news2016/1602/160208a.html>

本資料では、FASA のアーキテクチャについて説明するとともに、FASA アプリケーションを実装するための汎用化した入出力インタフェース (FASA アプリケーション API) についても説明をする。

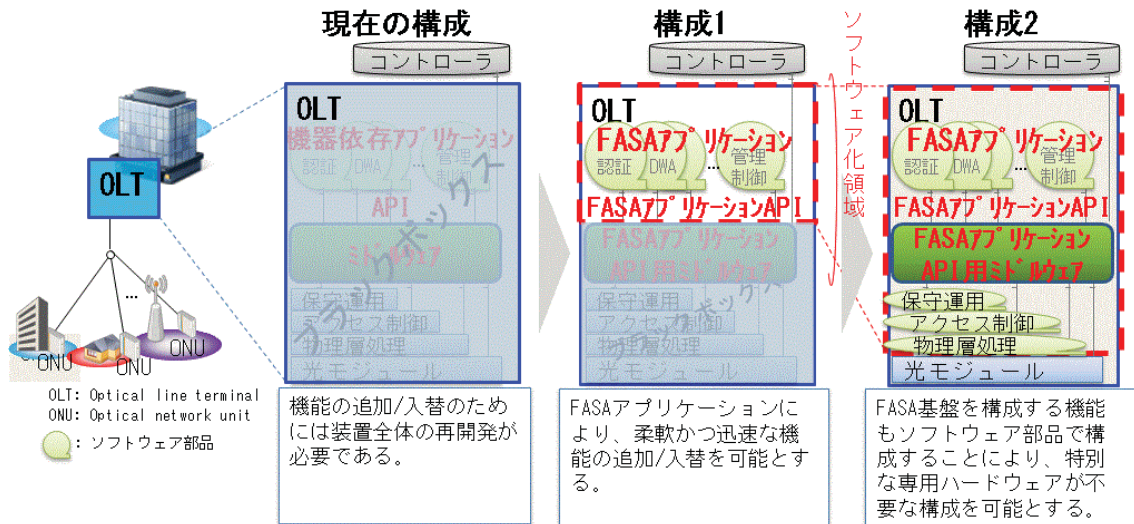


図 1. FASA による部品化の構成例

1. はじめに

1.1. 本資料の目的

本資料では、様々な通信事業者要件、様々なサービスやアプリケーションへの迅速な対応が可能となる新アクセスシステムアーキテクチャ FASA(Flexible Access System Architecture)と FASA アプリケーションを実装するための汎用化した入出力インターフェースである FASA アプリケーション API (Application Programming Interface) について述べる。

1.2. FASA 概要

図 1.2.-1 に、アクセスシステムの現状とあるべき姿を示す。

現在のシステム・装置構成では、サービス専用のシステム・装置として機能の実装がなされているため、機能の追加や入替にはシステム・装置全体の再開発が必要となる等制約が多く、また、保守運用についてもそれぞれの装置に対応する保守部品やスキルが必要となっている状況である。そのため、図に示すように、アクセスネットワーク装置の柔軟性や拡張性を向上し、様々な通信事業者要件及びサービスへの迅速な対応を可能とすることを目指す必要がある。

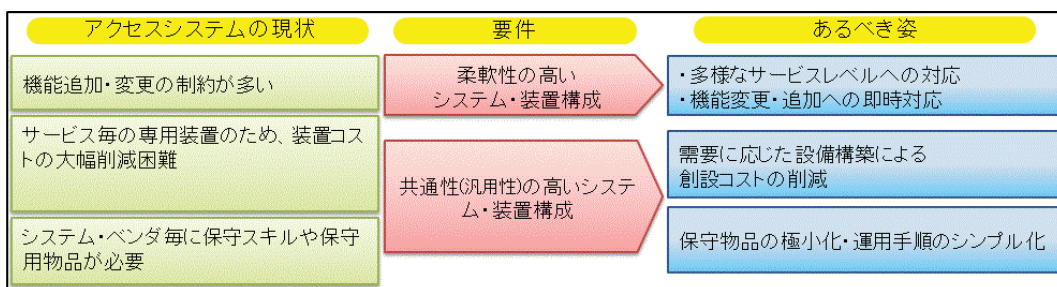


図 1.2.-1 アクセスシステムの現状とあるべき姿

FASA は、以下を実現する新たなアクセスシステムアーキテクチャ及びそのコンセプトである。

1. これまでのように、機能やサービスに特化した専用装置として開発するのではなく、アクセスネットワーク装置を構成する機能を部品化する。
2. サービス毎及び通信事業者毎に異なる機能は、入出力インターフェースを汎用化したソフトウェア部品で実現する。
3. ソフトウェア部品間の独立性を高めることで、ソフトウェア部品を入替可能な基盤上で動作させることによって、サービス品質を維持しながら、必要な機能をサービス要件に応じて柔軟かつ経済的に実現する。

FASA による部品化の概念図を図 1.2.-2 に示す。図に示すように、FASA に基づくアクセスネットワーク装置は FASA アプリケーションと FASA 基盤とから構成される。

「FASA アプリケーション」は、サービス毎あるいは通信事業者毎に異なる機能を、汎用化した入出カインタフェース (FASA アプリケーション API) を備えたソフトウェア部品で実現し、それらを入替可能としたものである。サービスに応じて、FASA アプリケーションを追加や入替することで、様々な要件のサービスを迅速かつ簡単に提供する。

「FASA 基盤」は、FASA アプリケーションに FASA アプリケーション API を提供するとともに、標準化されているなどの理由で、サービスや要求に応じた変更を行う必要のない機能を提供するアクセス装置の基盤的構成要素である。FASA 基盤では、処理性能等の要件に応じて、FASA 基盤を構成する各機能を、ハードウェアあるいはソフトウェアで実現する。具体的な FASA による部品化の構成例を、図 1.2.-3 に示す。

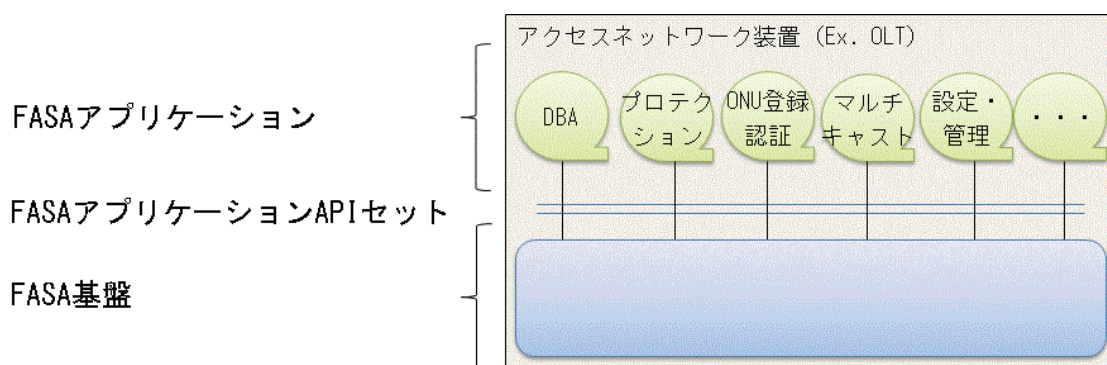


図 1.2.-2 FASA により実現する部品化の概念図

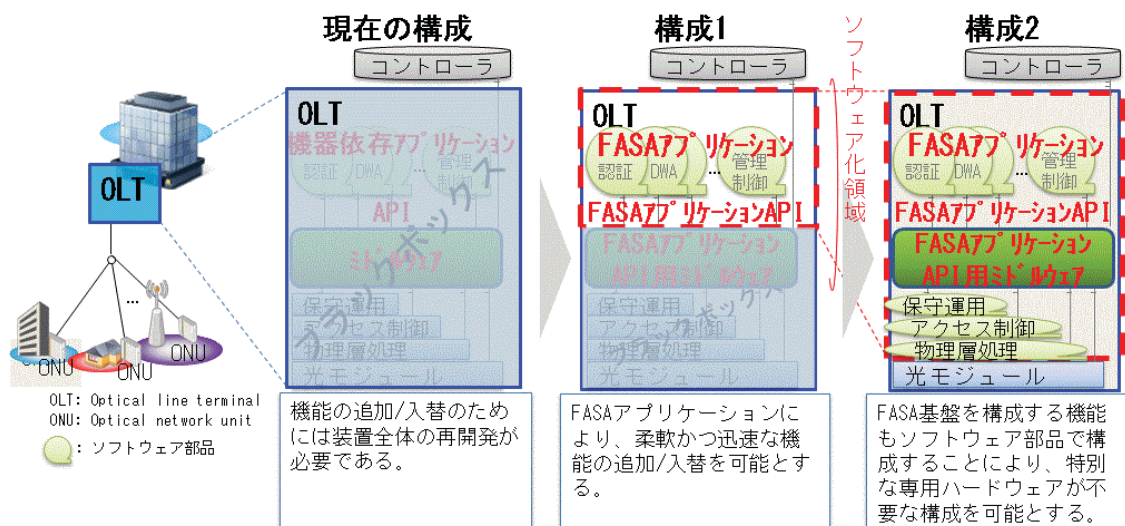


図 1.2.-3 FASA による部品化の構成例

構成 1 は、FASA アプリケーション API の上部 (FASA アプリケーション) をソフトウェア部品化の対象としたものであり、FASA 基盤を構成する各機能の部品化については対象外とし

ている。例えば本構成による NG-PON2 システムの場合、FASA 基盤にて標準化された NG-PON2 プロトコルの提供を行い、EPON など他の PON プロトコルへの変更については考慮しない。

構成 2 では、FASA 基盤を構成する各機能についても部品化を行い、ソフトウェア部品化の対象としている。

構成 1 と構成 2 のいずれの構成でも、FASA アプリケーション API は同一である。

FASA 基盤を汎用ハードウェアと外付ハードウェア部品に分割した例を図 1.2.-4 に示す。図に示すアクセスネットワーク装置は、3 つの部品カテゴリ、「(1)FASA アプリケーション」、「(2) 汎用ハードウェア」、「(3) 外付ハードウェア部品」から構成される。これらを組み合わせることにより必要な機能を迅速かつ簡単に提供する。「(2) 汎用ハードウェア」は、アクセスネットワーク装置に限らず汎用的な通信機能を共通部品として実装したものである。汎用ハードウェアによる共通部品化を図ることで、サービス要件に応じて装置を部材レベルから新規に開発する頻度を低減できる。また、共通部品を用いることにより、装置コスト削減や、保守物品種別等の削減による保守運用のシンプル化が進むことが期待できる。「(2) 汎用ハードウェア」は、例えば、汎用的なサーバやホワイトボックススイッチ等のアクセスシステム専用装置ではないハードウェアであり、当該ハードウェアに必要なファームウェア、OS 等のソフトウェア、「(1)FASA アプリケーション」のための FASA アプリケーション API 用ミドルウェアを搭載している。「(3) 外付ハードウェア部品」は、「(1)FASA アプリケーション」化や「(2) 汎用ハードウェア」上への実装が困難な、光送受信部等の機能を、「(2) 汎用ハードウェア」と分離して実装したものである。「(3) 外付ハードウェア部品」は、例えば光送受信部等の「(2) 汎用ハードウェア」以外のハードウェアである。「(3) 外付ハードウェア部品」として単純な光モジュールを用いて、メディアコンバータを実現する場合等においては、「(2) 汎用ハードウェア」に搭載される OS 等のソフトウェア、「(1)FASA アプリケーション」のための FASA アプリケーション API 用ミドルウェアを用いて動作することとなる。一方、「(3) 外付ハードウェア」に PON の MAC チップを搭載する場合等では、当該ハードウェアの動作に必要なファームウェア、OS 等のソフトウェア、「(1)FASA アプリケーション」のための FASA アプリケーション API 用ミドルウェアを「(3) 外付ハードウェア」に直接搭載する場合がある。いずれにせよ、サービス要件に応じて例えば伝送容量や伝送方式が異なる外付ハードウェア部品を入替し、それに応じたソフトウェア部品を FASA 基盤に具備すなわち FASA 基盤を構成するソフトウェア部品の入替を行うことで、同一の汎用ハードウェアを用いながら状況に応じた最適な伝送容量や伝送方式のアクセスネットワークの実現が可能である。

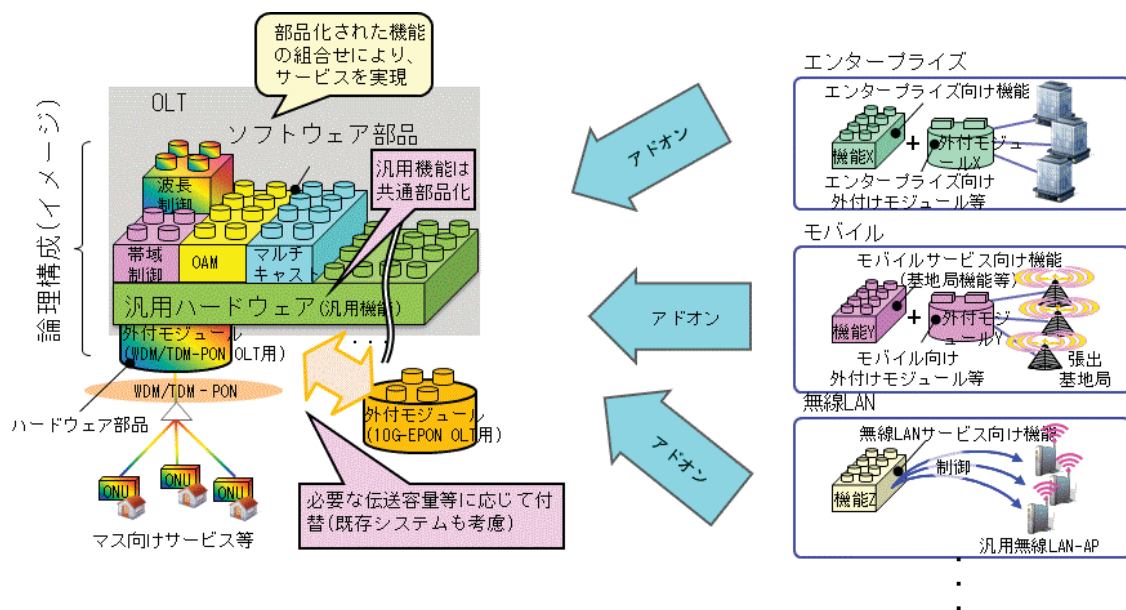


図 1.2.-4 FASA 基盤の部品化の考え方の例

1.3. 本資料における FASA アプリケーション API の規定範囲

本資料では、FASA を適用したアクセスシステムのユースケースと、ユースケースを実現するための具体的なアーキテクチャ、FASA アプリケーションとする機能及び FASA アプリケーション API について述べる。なお、現時点では対象とするアクセスシステムによっては、処理性能等の観点で PON の MAC チップ等の専用ハードウェアが必要となる場合があることから、FASA 基盤で提供する各機能の部品化(図 1.2.-3 の構成 2)については本資料の対象外とし、図 1.2.-3 の構成 1 の PON OLT を対象とする。

1.4. 用語と定義及び略語

FASA-API: FASA で用いる API の総称。

FASA アプリケーション: FASA アプリケーション API を使って実現された入替可能なソフトウェア部品。

FASA アプリケーション API: FASA アプリケーションと FASA アプリケーション API 用ミドルウェアミドルウェアを接続する API。

FASA アプリケーション API 用ミドルウェア: FASA 基盤のうち、FASA アプリケーションに対して FASA アプリケーション API を提供するソフトウェア。FASA アプリケーション API 用ミドルウェアは、FASA アプリケーション間及びその他の FASA 基盤の機能との通信のための手段を提供するとともに、FASA アプリケーションより下部の機能等の差異を吸収する。

FASA 基盤: FASA アプリケーションに FASA アプリケーション API を提供するとともに、標準化されているなどの理由で、サービスや要求に応じた変更を行う必要のない

機能を提供するアクセス装置の基盤的構成要素である。

FASA 基盤 API: FASA 基盤内のソフトウェア部品に用いる API。

外付ハードウェア部品: 汎用ハードウェア上への実装が困難な機能を、汎用ハードウェアと分離して実装したもの。

ソフトウェア部品: 必要な機能を交換可能な単位でソフトウェア化したもの。

汎用ハードウェア: 汎用的なサーバやホワイトボックススイッチのように、アクセスネットワークサービスに限らず汎用的に使えるハードウェア。

ACK: ACKnowledgement

ACPI: Advanced Configuration and Power Interface

API: Application Programming Interface

ASIC: Application Specific Integrated Circuit

B2B: Business-to-business

B2B2C: Business to Business to Consumer

BBU: Base Band Unit

BWMap: BandWidth Map

CLI: Command Line Interface

CT: Channel Termination

DBA: Dynamic Bandwidth Assignment

DWA: Dynamic Wavelength Assignment

DWBA: Dynamic Wavelength and Bandwidth Assignment

EPON: Ethernet Passive Optical Network

FASA: Flexible Access System Architecture

FBA: Fixed Bandwidth Assignment

FEC: Forward Error Correction

FTTH: Fiber to the Home

GEM: G-PON Encapsulation Method

G-PON: Gigabit-capable Passive Optical Network

GTC: G-PON Transmission Convergence

IEEE: The Institute of Electrical and Electronics Engineers

IF: Interface

IPv6: Internet Protocol Version 6

ITU-T: International Telecommunication Union Telecommunication Standardization Sector

L2: Layer 2

L3: Layer 3

LTE: Long Term Evolution
MAC: Media Access Control
MFH: Mobile Front Haul
MLD: Multicast Listener Discovery
NBI: Northbound Interface
NE-OpS: Network Element-Operation System
NG-PON2: Next Generation Passive Optical Network 2
NSR: Non-Status Reporting
OAM: Operation Administration and Maintenance
ODN: Optical Distribution Network
OLT: Optical Line Terminal
OMCI: ONU Management and Control Interface
ONU: Optical Network Unit
OS: Operating system
OSS: Operators Management System
OSU: Optical Subscriber Unit
PHY: Physical Layer
PLOAM: Physical Layer OAM
PON: Passive Optical Network
PS: Power Save
PTP: Precision Time Protocol
QoS: Quality of Service
RRH: Remote Radio Head
SBI: South Bound Interface
SD: State Diagram
SDK: Software Development Kit
SFC: Super Frame Counter
SNI: Service Node Interface
SNMP: Simple Network Management Protocol
SP 変換: Serial to Parallel 変換
SR: Status Reporting
SyncE: Synchronous Ethernet
TBD: To Be Determined
TC: Transmission Convergence
TDD: Time Division Duplex
TDM: Time Division Multiplexing

ToD: Time of Day
TRx: Transceiver
UE: User Equipment
UNI: User Network Interface
VLAN: Virtual LAN (Local Area Network)
XGEM: XG-PON Encapsulation Method
XGTC: XG-PON Transmission Convergence
XG-PON: 10 Gigabit Capable Passive Optical Network
WDM: Wavelength Division Multiplexing
WDM/TDM-PON: Wavelength Division Multiplexing and Time Division Multiplexing
Passive Optical Network

1.5. 参考文献

- [1] ITU-T G.989 シリーズ
- [2] IEEE std.802.3TM, 2015
- [3] IEEE std.1904.1, 2013
- [4] ITU-T G.supplement 51
- [5] T. Tashiro, et al., “A Novel DBA Scheme for TDM-PON based Mobile Fronthaul,” OFC2014, paper Tu3F.3, Mar. 2014.
- [6] D. Hisano, et. al., “Efficient Accommodation of Mobile Fronthaul and Secondary Services in a TDM-PON System with Wireless TDD Frame Monitor,” “IEEE ICC 2016, paper ONS1.1, May 2016.
- [7] T. Kobayashi, et al., “Bandwidth Allocation scheme based on Simple Statistical Traffic Analysis for TDM-PON based Mobile Fronthaul,” OFC 2016, paper W3C.7.

2. 参照アーキテクチャ

2.1. 論理モデル

論理モデルを図 2.1.-1 に示す。モデルは OLT の例であり、コントローラと外部装置は OLT に含まれないが、FASA アプリケーション API との通信を例示するために記載する。論理モデルは、FASA アプリケーションと、FASA アプリケーションに FASA アプリケーション API を提供する FASA 基盤とから構成される。FASA 基盤は FASA アプリケーション用ミドルウェアを含む。

FASA アプリケーション API 用ミドルウェアは、FASA 基盤を構成するハードウェアやソフトウェアのベンダや方式の違いを吸収する。FASA アプリケーション API 用ミドルウェア上にベンダや方式に依存しない FASA アプリケーション API セットを規定し、FASA アプリケーションの入替により、サービス毎あるいは通信事業者毎に必要な機能を実現する。FASA ア

アプリケーション間の通信やコントローラ等による設定管理は FASA アプリケーション API 用ミドルウェアを介して行う。

FASA アプリケーション API セットは、FASA アプリケーションで利用する共通の API 群であり、FASA アプリケーション毎に必要な API を API セットから選択して利用する。

コントローラは、NE-OpS 等の OLT の設定管理システムであり、コントローラからの制御信号は FASA アプリケーション API を介して SBI アプリと通信する。SBI アプリと通信する制御信号は、FASA アプリケーション API を介して制御管理アプリで終端される。

外部装置は、例えばモバイルシステムの BBU や他 OLT であり、FASA アプリケーションと FASA アプリケーション API を介して通信する外部の装置である。図では、外部装置 (BBU) が DBA アプリと通信している。

図では、FASA アプリケーション API 用ミドルウェアより下部の機能との通信、アプリ間通信、アプリ間通信の内の設定管理アプリとその他アプリとの通信を、それぞれ赤色、緑色、オレンジ色の矢印で示す。

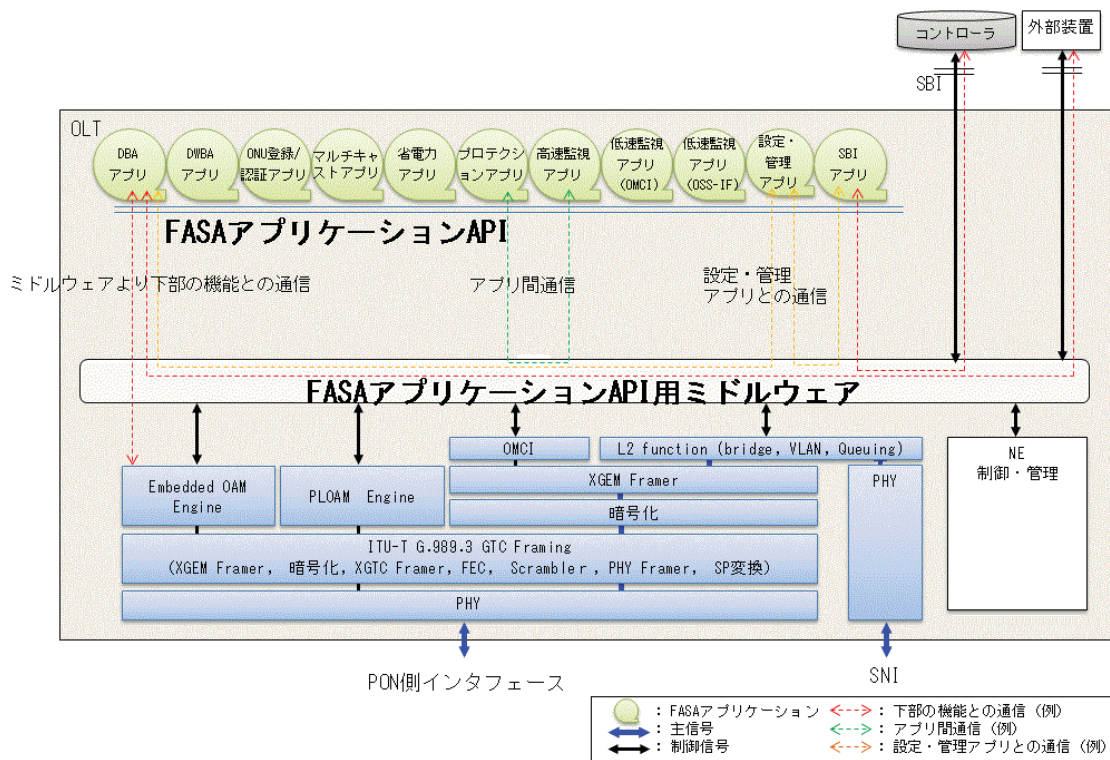


図 2.1.-1 FASA の論理モデル

2.2. システム構成例

NG-PON2 OLT を例にとり、FASA に基づくアクセスネットワーク装置の構成例を図 2.2.-1 に示す。図は、FASA 基盤を複数のハードウェア (NG-PON2 ボックス、ホワイトボックススイ

ッチ)で構成した例を示している。NG-PON2 ボックスとホワイトボックススイッチとは、Ethernet 等の標準的なプロトコルで接続した。機能の追加や入替は、FASA 基盤の FASA アプリケーション API 上への FASA アプリケーションの追加や入替により行う。FASA アプリケーション API 用ミドルウェアは、ベンダや方式の差異によるハードウェア及びソフトウェアの違いを吸収し、FASA アプリケーション API を提供するためのソフトウェアである。

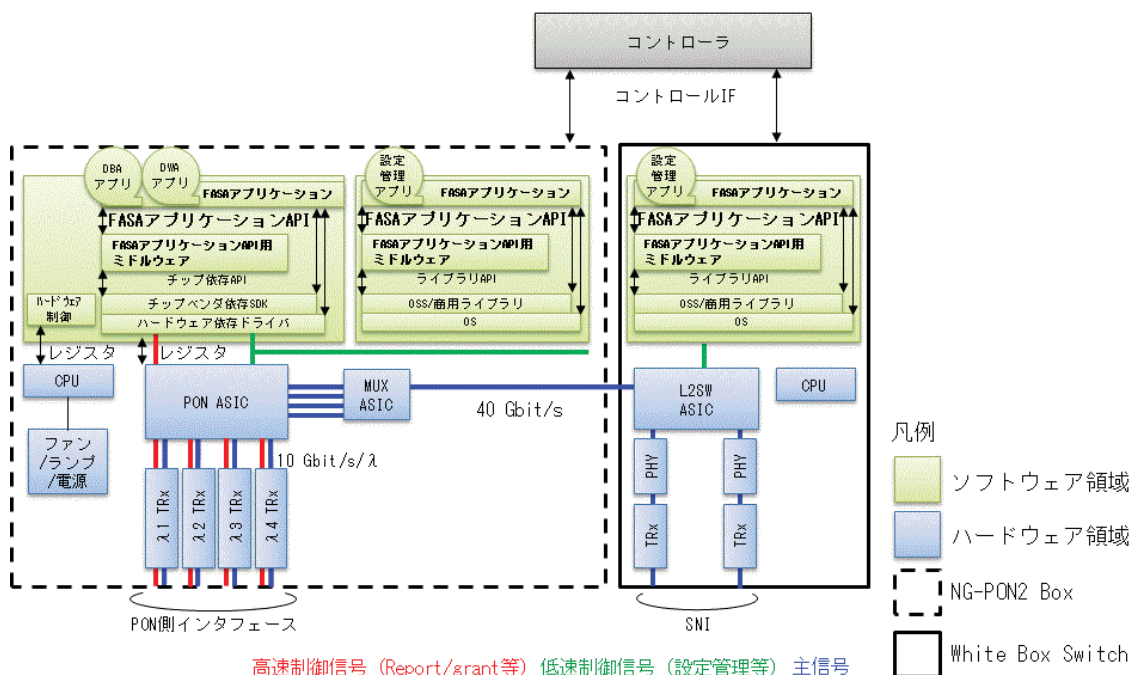


図 2.2.-1 FASA に基づくアクセスネットワーク装置 (NG-PON2 OLT) 構成例

3. ユースケース

FASA のユースケースとして、FASA アプリケーション入替によるマルチサービス収容と、FASA アプリケーション入替による通信事業者独自仕様の実現を示す。

前者は、FASA アプリケーションを入替することで、モバイル、マス、ビジネス等の要求条件が、大きく異なるサービスを PON システムに収容する。3.1. では、その代表的な例である DBA のユースケースについて述べる。

後者は、通信事業者毎に使い方の異なる機能に対応するために、全ての組み合わせを予め実装しておくのではなく、必要に応じて、FASA アプリケーション入替によって所望の機能を備えたアクセスネットワーク装置を提供する。3.2. 及び 3.3. では、その代表的な例である省電力化及びプロテクションのユースケースについて述べる。図 3.-1 に、FASA アプリケーション入替による独自仕様の入替イメージを示す。

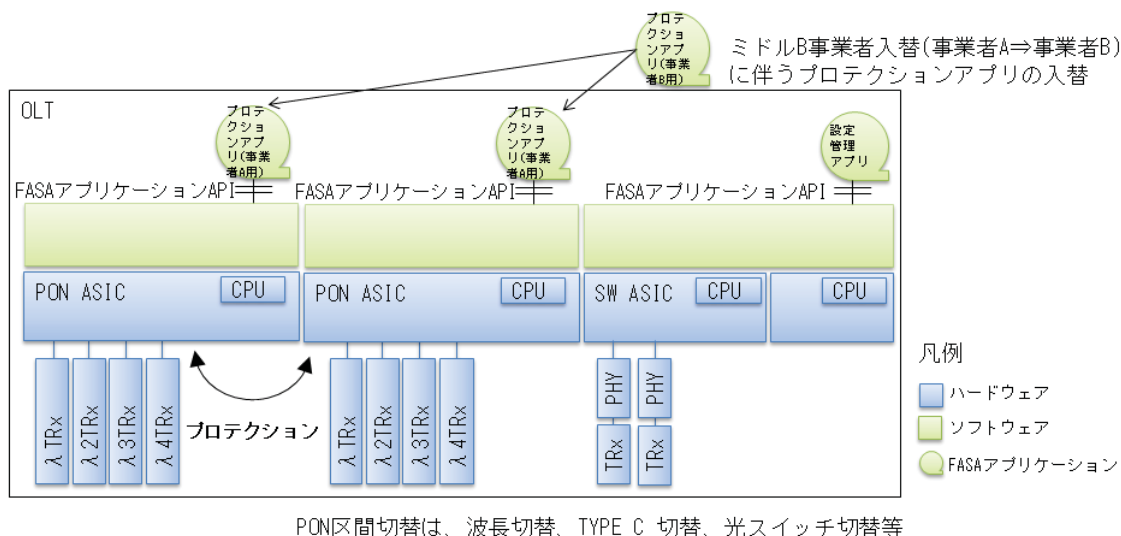


図 3.-1 FASA アプリケーション入替による独自仕様の入替

3.1. DBA の入替によるマルチサービス収容の実現

現在 FTTH に主に利用されている PON システムを用いて、モバイル、マス、ビジネス等のマルチサービスを効率的に提供することが期待されている。しかし、これらのサービス毎の要求条件は大きく異なっている。特に、モバイルフロントホール(MFH)の主信号に対する最大許容遅延は、例えば従来のインターネットアクセスサービスに比べ厳しく規定されている。そのため、PON の上りの帯域を割当する DBA は、厳しい遅延規定を満たす必要がある。また、MFH への PON の適用に関しては、第五世代モバイル通信などの今後の技術の進展や規格化の進展等により要件が変化することが容易に想定され、その変化にタイムリに追従したシステム開発が必要である。FASA において、FASA アプリケーション入替により、各サービスに応じた DBA をタイムリに提供する。

3.1. では、DBA のユースケースの網羅的な分析から、将来のサービス追加や入替に対応可能な API を規定する。

3.1.1. MFH 向け DBA

DBA は、各 ONU の上り帯域(送信可能タイムスロット)を割当する処理であり、ONU からの申告をもとに各 ONU に動的に帯域を割当する SR-DBA と、ONU からの申告を用いない NSR-DBA に分類できる。SR-DBA では、ONU の申告に応じた帯域利用効率の高い割当ができる一方で、OLT-ONU 間の制御信号の往復に時間を要するため、MFH の要件を満たすことが困難である。そこで、外部装置から帯域割当に必要な情報を取得する光モバイル連携 DBA と、トラフィック予測に基づき帯域割当する NSR-DBA を提案している[5-7]。

3.1.1.1. MFH 向け光モバイル連携 DBA

MFH 向け光モバイル連携 DBA の構成とシーケンスを図 3.1.1.1.-1 に示す。MFH 向け光モバイル連携 DBA では、BBU と OLT との連携によって、MFH の遅延規定を満たす帯域割当を実現する。具体的には、基地局 (RRH) からの上りパケット到着に基づく ONU からの帯域割当要求の代わりに、各 UE に対して BBU が生成する上り送信制御情報相当の無線リソース情報を用いて、帯域割当情報(各 ONU の上り帯域とタイミング)を DBA アプリあるいは外部装置で算出して上り帯域の割当を行う[5]。この動作により、上り信号が ONU に到着してから PON 区間に ONU から信号が出力されるまでの待ち時間をなくし、遅延を低減する。本 DBA では、外部装置との連携を可能とする API が必要となる。

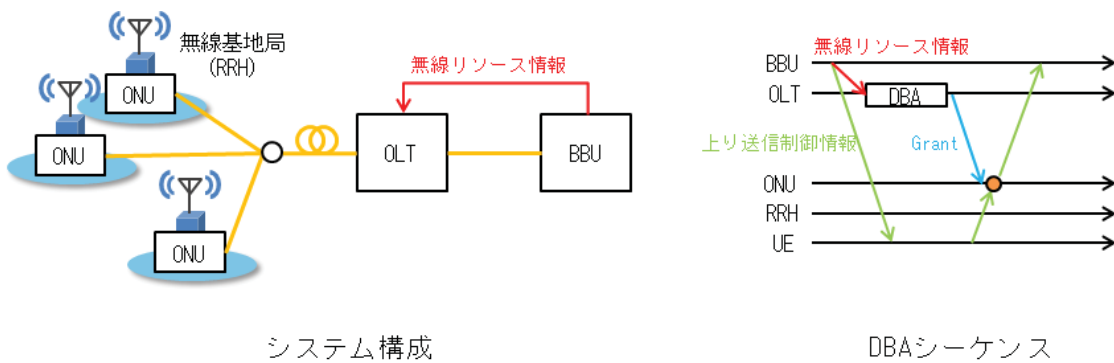


図 3.1.1.1.-1 MFH 向け光モバイル連携 DBA

3.1.1.2. MFH 向け NSR-DBA

図 3.1.1.2.-1 に 2 種類の MFH 向け NSR-DBA の例の概要を示す。これらは、固定帯域割当 (FBA) をベースとした NSR-DBA により、SR-DBA で生じる制御信号の往復遅延を削減し、MFH の遅延規定を満たす帯域割当を実現する。具体的には、トラフィックの統計データ及びトラフィックパターンに基づき NSR-DBA を適用することで、FBA で生じる過剰な割当帯域を削減しつつ、マス向けシステムとの共用等により、システム全体の高効率化を可能にする。

一つ目の NSR-DBA は、TDD 方式のモバイルシステムの TDD 周期をトラフィックの統計データから推定し、TDD 周期に応じて帯域割当を行う[6]。本 DBA では、TDD 周期を推定するために十分なサンプリング速度でトラフィック統計情報を利用可能とする API が必要となる。

二つ目の NSR-DBA は、トラフィックの変動が 1 日~1 週間程度の時間軸で観測すると周期的になっていることに着目し、トラフィック統計情報をもとに次のトラフィック周期に必要な帯域を推定して割当を行う[7]。本 DBA では、長期に渡るトラフィック統計データを利用可能とする API が必要となる。

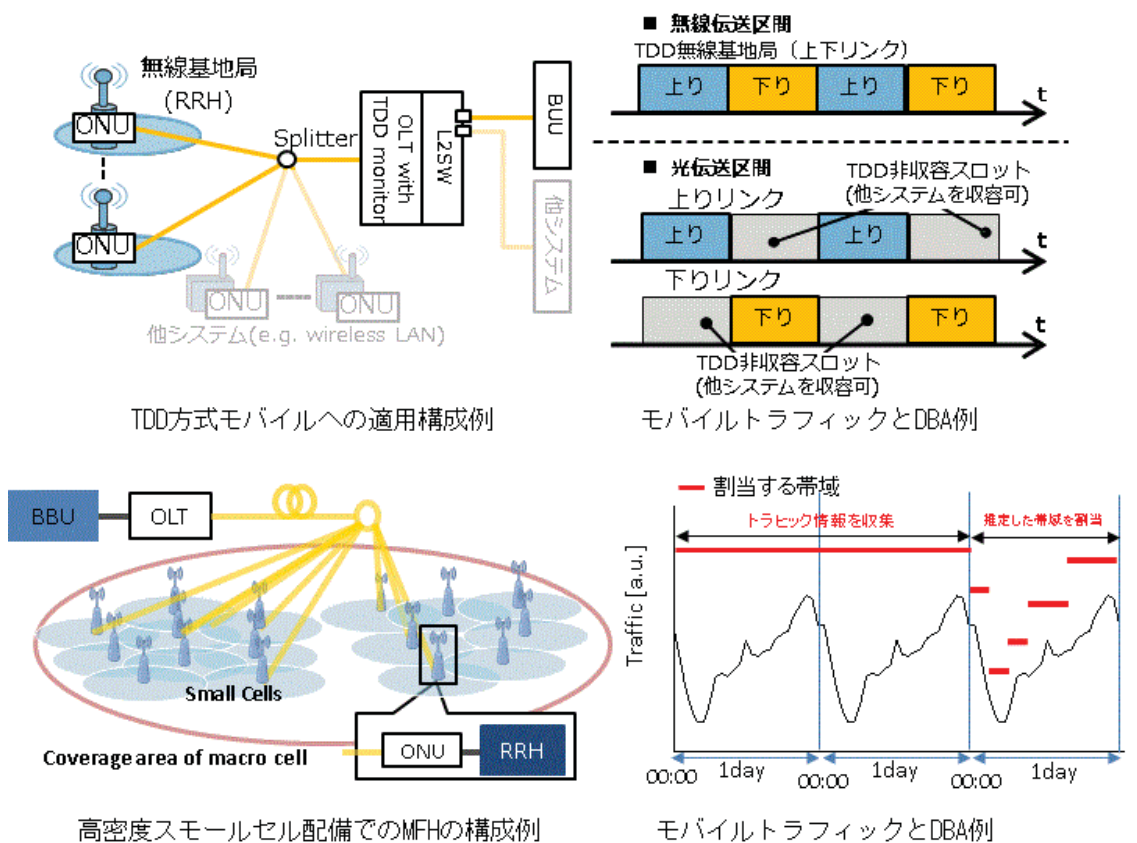


図 3.1.1.2.-1 MFH 向け NSR-DBA

3.1.2. マス向け SR-DBA

図 3.1.2.-1 に、マス向け SR-DBA の構成とシーケンスを示す。マス向け SR-DBA では、各 ONU の申告から抽出した要求量をもとに、各 ONU に動的に帯域割当を行う。よって、本 DBA では、各 ONU の申告から要求量を抽出する API が必要である。

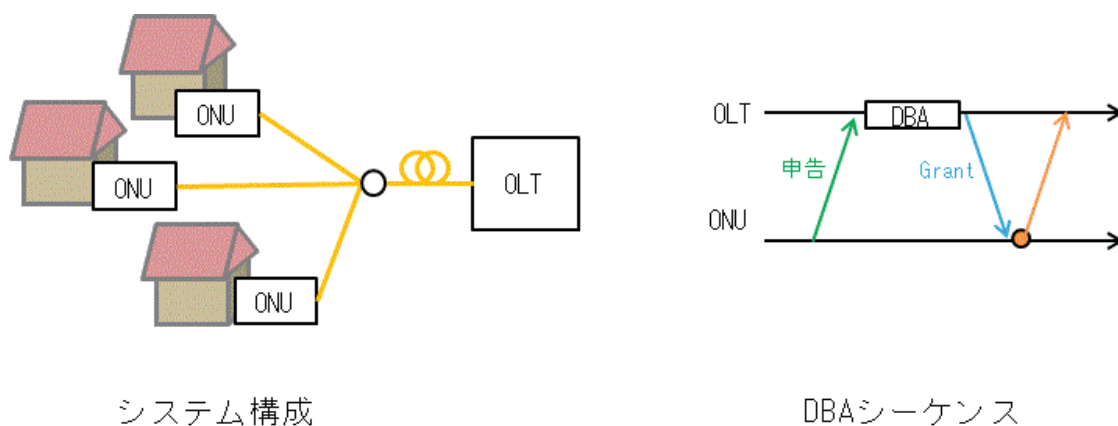


図 3.1.2.-1 マス向け SR-DBA

3.1.3. ビジネス向け DBA

TBD

3.1.4. DBA の API と機能ブロック

以上から、各ユースケースを実現するための DBA の API と機能ブロック図を図 3.1.4.-1 に示す。また、図 3.1.4.-1 の各機能部(方針決定部、割当計算部、連携制御部、トラフィックモニタ、申告処理部、grant 処理部)の定義を、図 3.1.4.-2 に示す。

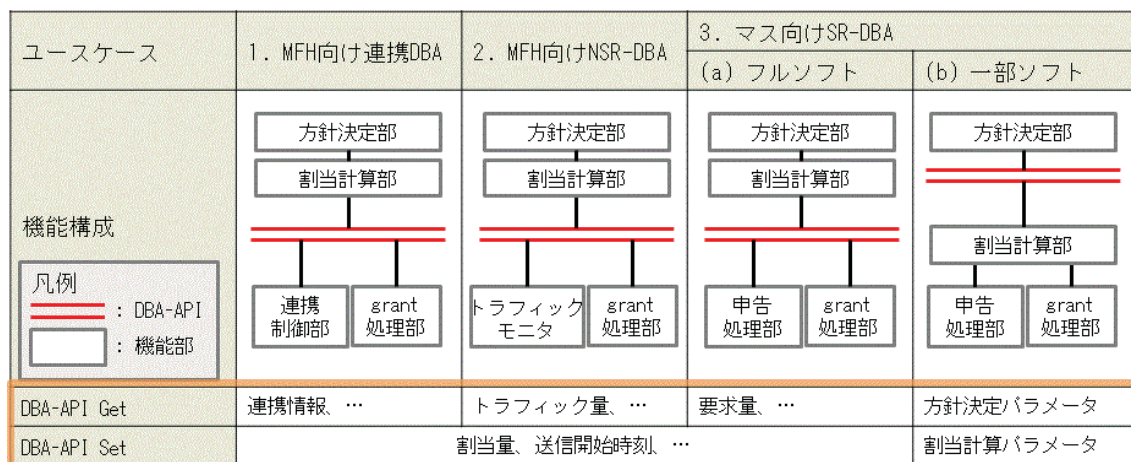


図 3.1.4.-1 DBA の API と機能ブロック図

図 3.1.4.-1 のユースケース 1 及び 2 は、外部装置の連携等の DBA を柔軟に実装するために、DBA 全体(方針決定部及び割当計算部)を FASA アプリケーションとして実装する場合の構成と API の位置を示す。また図 3.1.4.-1 のユースケース 3 のマス向け SR-DBA は、DBA 全体を FASA アプリケーションとして実装する「(a)DBA 機能のフルソフト化」に加え、従来の PON の MAC チップ等と同様な構成として DBA の一部(方針決定部)のみを実装する「(b)DBA 機能の一部ソフト化」の場合の構成と API の位置を示す。

以上のユースケースに対応するために、FASA 基盤は、PON の標準で規定されている申告処理機能、grant 処理機能に加えて、連携制御機能、トラフィックモニタ機能を有し、連携情報取得、トラフィック量取得、要求量取得、割当計算パラメータ取得、割当量設定、送信開始時刻設定、方針決定パラメータ設定の API を提供する必要がある。

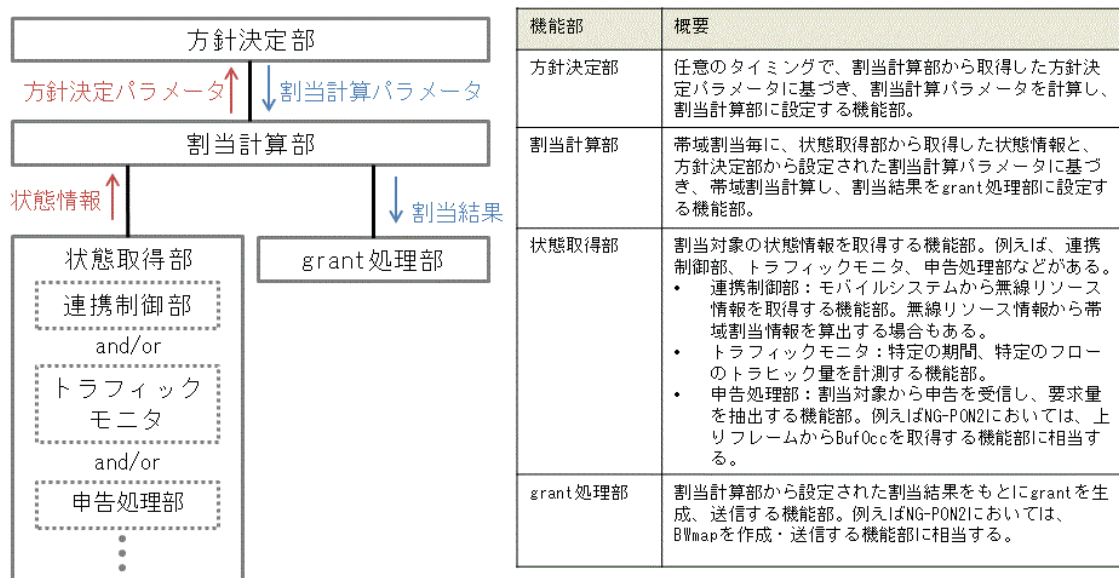


図 3.1.4.-2 機能部の定義

3.2.1. 省電力機能の入替による通信事業者独自仕様の実現

TBD

3.2.2. プロテクション機能の入替による通信事業者独自仕様の実現

通信事業者要件として、アクセスシステムに対しても高い可用性が必要となる。装置故障時の対応のみならず、アクセスネットワーク装置のソフトウェア更新や処理性能が向上した後継ハードウェアへ移行する際も、サービス停止なく移行する必要があるため、プロテクション機能が重要となる。

通信事業者毎に、アクセスネットワーク装置と接続するネットワークの構成が異なることから、異なる方式や異なるポリシーでプロテクションを行うことが想定されるため、FASA ではプロテクション用 API を規定する。

4. アクセスシステムの主要機能と FASA アプリケーション化の対象

アクセスシステムの主要機能と FASA アプリケーション化の対象を表 4.-1 に示す。まず、アクセスシステムの主要機能について説明する。

PON 主信号処理機能は、ONU との間で送受信を行う主信号を処理する機能群であり、PON フレームの生成分離や前方誤り訂正 (FEC: Forward Error Correction) 等の機能を含む。

PON アクセス制御機能は前述の主信号送受信を行うための制御機能群であり、動的帯域割当や ONU の登録認証等の機能を含む。

L2 主信号処理機能は、PON 側ポートと SNI 側ポートとの間で主信号を転送し、処理する機能群であり、MAC アドレス学習や VLAN 制御、優先制御やトラフィックモニタ等の機能を

含む。

保守運用機能は、アクセス装置によってサービスを円滑に保守運用するための機能群であり、ONU や OLT (OSU 及びスイッチ) の装置設定を実施する機能や、ソフトウェアの更新、装置やサービスの管理、各種機能が正常に動作しているかを監視する機能、異常発生時に能動的に警報を発出する機能、異常発生時の範囲や原因を調査するための試験機能等を含む。また、保守運用機能は、多数のアクセス装置を管理する保守運用システムと接続され、リモートからも円滑な保守運用を実現する。

PON マルチキャスト機能は、SNI 側から受信したマルチキャストストリームを適切なユーザに転送する機能群であり、マルチキャストストリームの識別や振分、ONU のフィルタ設定を実施する機能を含む。

省電力制御機能は、ONU や OLT の電力消費を削減するための機能群であり、標準化で規定されている省電力化機能に加え、トラフィックモニタとの連携によってサービスへの影響を最小限に抑えながら、最大限の省電力効果を得るための機能を含む。

周波数/時刻同期機能は、ONU 配下の装置に正確な周波数同期や時刻同期を提供するための機能群であり、自身のリアルタイムクロックを上位装置に従属同期させる機能や、PON フレームを用いて ONU に時刻情報を通知する機能を含む。

プロテクション機能は、スイッチ間や OSU 間等、複数のハードウェアで冗長をとった構成において、障害検知時に現用系から予備系への切替や引継を実施してサービスを継続するための機能群であり、切替トリガの検出や切替処理の実施といった機能を含む。また、プロテクション機能は障害検知時や手動での切替時に、サービスを全面停止せず縮退運転で動作させ続けるための機能を提供する。

FASA ホワイトペーパー Ver. 1.0

表 4.-1 アクセスシステムの主要機能と FASA アプリケーション化の対象

機能群	機能	FASAアプリケーションAPI利用による拡張例	FASAアプリケーション/FASA基盤
PON主信号処理機能	基本機能	標準準拠のフレーム/ヘッダの生成・分離、フラグメント処理、スクランブラ、FEC、暗号化。	基盤
PONアクセス制御機能	ONU登録/認証	独自認証手順(不正利用抑止)	アプリ
	DBA	標準準拠のDBAフレーム処理(フレームヘッダの読み書き等)	基盤
		独自QoS規定(遅延、公平性、帯域利用率等)への対応	アプリ
		サービス要求(例: モバイル提供のための低遅延要求)への対応	アプリ
		プログラムの暗号化への対応	アプリ
	DWA	標準準拠の波長切り替えシーケンス(ITU-T G.989.3)	基盤
独自QoS規定(遅延、公平性等)への対応		アプリ	
独自保守運用システムとの整合性確保		アプリ	
トラフィックモニタとの連動による特性向上		アプリ	
DoS対策	独自フィルタ(特定のフィールドとのマッチ、かつ一定以上の流量等)の適用	アプリ	
L2主信号処理機能	SNIポート	ミラーリングポートの具備、マルチキャストポートの具備、コネクタ種別、送受信フレーム(Ethernet)、オートネゴシエーション、伝送媒体の指定	基盤
	ブリッジ機能	標準準拠のブリッジ処理(MACアドレス学習、IEEE 802.1Q VLAN tag処理等)	基盤
	トラフィックモニタ	トラフィックモニタの具備(具体的な項目は仕様が必要)	基盤
	集線	ポリリング機能の具備、メータ/マーカの具備	基盤
		独自QoS規定(遅延、パケットロス、公平性等)への対応	アプリ
	優先制御	標準準拠のキュー/スケジューラ(SPO)の具備	基盤
		独自QoS規定(遅延、パケットロス、公平性等)への対応	アプリ
VLAN管理	独自保守運用システム(PON区間のVLANの使い方等)との整合性確保	アプリ	
保守運用機能	保守運用フレーム処理	標準準拠の保守運用フレーム(DMCI, PLOAM, OAM)処理	基盤
	保守運用ポート/監視制御ポート	コネクタ種別、送受信フレーム(Ethernet)、オートネゴシエーション、伝送媒体の指定	基盤
	SBI	コントローラとの制御信号のやりとり、ONUやOLTが他の伝送装置やサーバと連携を取るためのタイミング制御機能	アプリ
	設定	CLIの具備	基盤
		FASAアプリケーション追加、変更時への対応(更新手段の提供等)	アプリ
		独自保守運用システムとの整合性(保守運用システムとの接続プロトコル、交換パラメータ等。例: ユーザの契約サービスの取得)確保	アプリ
	管理	独自ソフトウェア更新方式(転送方式、更新タイミング、更新処理時間、断時間の許容値等)への対応	アプリ
		独自保守運用システムとの整合性(ONUの設定等)確保	アプリ
		独自ログ保存手段(保存エントリ数、形式等)	アプリ
	監視制御	SNMPエージェント具備	基盤
Ether-OAM具備		基盤	
独自保守運用システムとの整合性確保(警報通知ポートの設定、警報通知用プロトコル等)		アプリ	
試験	独自の監視項目、独自の警報発出条件への対応	アプリ	
	独自試験(往復遅延時間測定)への対応等	アプリ	
PONマルチキャスト機能	IPマルチキャスト	マルチキャストフレーム(IGMPやMLD)処理	基盤
	フィルタ設定	独自フィルタ(VLAN/MACアドレスを用いる方式等)の適用	アプリ
	マルチキャストプロキシ	独自QoS規定(Zapping性能、信頼性等)	アプリ
	独自サービス(映像サービス等)との整合性確保等	アプリ	
省電力制御機能	ONU省電力化	標準準拠のシーケンス(ITU-T G.989.3やIEEE 1904.11)準拠	基盤
		独自QoS規定(遅延時間等)への対応	アプリ
	OLT省電力化	OLT省電力化に必要な状態遷移の具備	基盤
		独自QoS規定(遅延時間等)への対応	アプリ
周波数/時刻同期機能	同期手段	標準準拠(IEEE 1588等)の同期方式	基盤
		独自サービスとの整合性確保	アプリ
プロテクション機能	プロテクション	プロテクションに必要な状態遷移の具備(仕様化も必要)	基盤
		独自QoS規定(断時間等)への対応	アプリ
		複数の冗長構成(SW冗長、OSU冗長、1+1冗長、N:1冗長等)への対応	アプリ

次に、各機能を FASA アプリケーションとして実装するか、あるいは FASA 基盤上で実装するかの考え方と例について説明する。

表 4.-1 に示した機能のうち、サービスによって機能変更が必要になるものや通信事業者

独自の要件を満たすために拡張すべき機能を FASA アプリケーションとして実現する。一方、標準化等で規定されているため拡張の余地が少ない機能は FASA 基盤上に実装される。

表 4. -1 では、例えば、PON 主信号処理機能を FASA 基盤として実現することを示している。ITU-T G. 989 シリーズに準拠した 40 Gbit/s 級のアクセス装置を実現するには、フレームフォーマットや、フレームの暗号化、前方誤り訂正 (FEC) 機能といった基本的な PON 主信号処理機能は標準に従って実装する必要がある。また、こうした基本機能はサービスによらず共通であるため、FASA 基盤上に実装される。

別の例として、表 4. -1 では、PON アクセス制御機能に含まれる DBA 機能の「サービス要求への対応」を FASA アプリケーションとして実現することを示している。3 章で述べたように、提供するサービスによって、低遅延性を提供するケースや効率良く多数のユーザに帯域を割り当てるケースが存在する。サービス毎に異なる要求を満たすため、帯域割当の手順やポリシーを FASA アプリケーションとして、標準的な処理 (標準で規定されている、Bwmap フォーマットへの変換等) からは分離することが望ましい。

また、提供するサービスの対象が同じマス向けであっても、通信事業者によってヘビーユーザへの対応方針が異なる等、公平性のポリシーが異なることが考えられる。具体的には、例えば PON 単位といった粒度の小さい公平制御を必要とする通信事業者は DBA のアプリケーション内部でも公平制御を行い、アクセス装置単位といった大きい粒度でのみ公平制御を行う通信事業者は集線機能を用いることで、それぞれの QoS 規定を満たすことを想定している。

このように、FASA では異なる要求を FASA アプリケーションの入替によって実現するため、FASA アプリケーション入替の手段が必要となるが、入替手段として何を採用するかは、通信事業者や運用によって異なる。例えば、通信事業者が使用している既存の保守運用システムがソフトウェア更新に TFTP (Trivial File Transfer Protocol) を用いる場合は TFTP を備え、保守運用システムの外部から SFTP (SSH File Transfer Protocol) を用いて更新する場合は SFTP を備える。また、今後、装置とコントローラ間のインタフェースに関して標準化の議論が進展すると想定しており、標準化の進展に追従したインタフェースの追加や変更についても考慮する必要がある。このため、表 4. -1 ではアクセス装置が接続する他システムやその運用に合わせてカスタマイズが必要となる機能も FASA アプリケーションとして実現する。

また、FASA では、FASA 基盤全体を完全二重化して行うプロテクションに限らず、FASA 基盤の一部のみで行うプロテクションについても想定する。例えば FASA 基盤が、光スイッチを備えて PON プロテクションに対応する場合や、一つの PON に対して複数波長を備えて波長プロテクションに対応する場合、スイッチのみを二重化する場合、あるいはこれらを組み合わせた場合等、複数の冗長構成が考えられる。プロテクション機能を FASA アプリケーションとして実装することで、期待する冗長構成に対応でき、また該当箇所を再利用することで、容易に多様な冗長構成にも対応できる。

5. API 規定

API は、追加や入替する FASA アプリケーションによらずに、共通的な API セットとし、共通的な API セットから各 FASA アプリケーションで利用する API を選択する。以降では、ITU-T 勧告に基づく PON システムを中心に記述するが、IEEE 標準の PON や PON 以外のアクセスシステムについても対象である。

5.1. PON 主信号処理機能

TBD

5.2. PON アクセス制御機能

5.2.1. DBA

表 5.2.1.-1 に、DBA の各ユースケース (MFH 向け光モバイル連携 DBA、MFH 向け NSR-DBA、マス向け SR-DBA (DBA 機能のフルソフト化)、マス向け SR-DBA (DBA 機能の一部ソフト化)) の API リストを示す。表の「ユースケース」列は、ユースケースと API の対応を示す。「ユースケース」の「1」「2」「3a」「3b」の各列にチェックが入った API が、それぞれ MFH 向け光モバイル連携 DBA、MFH 向け NSR-DBA、マス向け SR-DBA (DBA 機能のフルソフト化)、マス向け SR-DBA (DBA 機能の一部ソフト化) で利用される。なお、本資料では、イベント駆動型の DBA プログラムを想定する。

API は、追加や入替する DBA の FASA アプリケーション (DBA アプリ) によらずに共通的な API セットとし、共通的な API セットから各 DBA アプリで利用する API を選択する。例えば、「get_AllocationDbConfig」「set_GrantSize」「set_GrantSizeAndStartTime」は DBA アプリで共通に利用し、「get_CooperationConfig」「get_DataSize」「get_Traffic」「get_AllocUnitTraffic」「get_CalcConfig」は特定の DBA アプリが利用する。共通に利用する「get_AllocationDbConfig」は、応答データから、それぞれの DBA アプリに必要な情報を取得する。送信開始時刻を指定しない「set_GrantSize」は FASA 基盤にて、送信開始時刻を指定する「set_GrantSizeAndStartTime」は FASA アプリケーションにて、それぞれ送信時間開始時刻を決定する。

表の「割当対象」は、DBA で帯域割当の対象となる論理パスを示す。割当対象番号は、割当対象のインデックスである。コンフィグ番号は、設定パラメータのインデックスである。モニタ番号は、トラフィックモニタの計測対象のインデックスである。方針決定用パラメータは、方針決定部の計算で用いられるパラメータである。割当計算用パラメータは、割当計算部の計算で用いられるパラメータである。なお、表中の〈詳細未定〉と記載した項目、インデックスと設定パラメータの対応、方針決定用パラメータの内容及び割当計算用パラメータの内容は今後の検討事項である。

表 5.2.1.-1 DBA の API

API	入力	出力	概要	ユースケース			
				1	2	3a	3b
on_Init	(なし)	(なし)	初期化時に呼出されるコールバック関数。	✓	✓	✓	✓
on_Process	(なし)	(なし)	周期的に、あるいは特定の条件を満たした際に(不定期的に)呼出されるコールバック関数。	✓	✓	✓	✓
on_ConfigUpdated	(なし)	<詳細未定>更新された設定情報	設定情報が更新された際に呼出されるコールバック関数。	✓	✓	✓	✓
on_Killed	(なし)	(なし)	DBAプログラムがkillされた際に呼出されるコールバック関数。(例えば、DBAアプリの入替時の利用を想定)	✓	✓	✓	✓
get_DbaConfig	コンフィグ番号	<詳細未定>設定値	DBAコンフィグ情報を取得する関数。(例えば、DBA周期、Quiet Window Size等を想定)	✓	✓	✓	✓
get_AllocUnitDbaConfig	割当対象番号、コンフィグ番号	<詳細未定>設定値	当該割当対象のDBAコンフィグ情報を取得する関数。(例えば最大帯域、保証帯域等を想定)	✓	✓	✓	✓
get_CooperationConfig	コンフィグ番号	<詳細未定>設定値	連携制御部のコンフィグ情報を取得する関数。(例えば連携制御部の動作周期や動作モード、連携先のシステム名、バージョン等を想定)	✓			
get_TrafficMonitorConfig	コンフィグ番号	<詳細未定>設定値	トラフィックモニタのコンフィグ情報を取得する関数。(例えばトラフィックモニタの動作モード等を想定)	✓			
get_CalcConfig	コンフィグ番号	<詳細未定>設定値	割当計算部のコンフィグ情報を取得する関数。(例えば割当計算部の動作モード等を想定)				✓
get_ReportConfig	コンフィグ番号	<詳細未定>設定値	申告処理部のコンフィグ情報を取得する関数。(例えば申告処理部の動作モード等を想定)			✓	✓
get_GrantConfig	コンフィグ番号	<詳細未定>設定値	grant処理部のコンフィグ情報を取得する関数。(例えばgrant処理部の動作モード等を想定)	✓	✓	✓	✓
set_DbaConfig	コンフィグ番号、<詳細未定>設定値	(なし)	DBAコンフィグ情報を設定する関数(例えばDBA周期、Quiet Window Size等を想定)	✓	✓	✓	✓
set_AllocUnitDbaConfig	割当対象番号、コンフィグ番号、<詳細未定>設定値	(なし)	当該割当対象のDBAコンフィグ情報を設定する関数。(例えば最大帯域、保証帯域等を想定)	✓	✓	✓	✓
set_CooperationConfig	コンフィグ番号、<詳細未定>設定値	(なし)	連携制御部のコンフィグ情報を設定する関数。(例えば連携制御部の動作周期や動作モード、連携先のシステム名、バージョン等を想定)	✓			
set_TrafficMonitorConfig	コンフィグ番号、<詳細未定>設定値	(なし)	トラフィックモニタのコンフィグ情報を設定する関数。(例えばトラフィックモニタの動作モード等を想定)	✓			
set_CalcConfig	コンフィグ番号、<詳細未定>設定値	(なし)	割当計算部のコンフィグ情報を設定する関数。(例えば割当計算部の動作モード等を想定)				✓
set_ReportConfig	コンフィグ番号、<詳細未定>設定値	(なし)	申告処理部のコンフィグ情報を設定する関数。(例えば申告処理部の動作モード等を想定)			✓	✓
set_GrantConfig	コンフィグ番号、<詳細未定>設定値	(なし)	grant処理部のコンフィグ情報を設定する関数。(例えばgrant処理部の動作モード等を想定)	✓	✓	✓	✓
get_DataSize	時刻、割当対象番号	要求量	当該割当対象の要求量を連携制御部から取得する関数。	✓			
get_TimeAndDataSize	割当対象番号	時刻、要求量	当該割当対象のONU到着時刻と要求量を連携制御部から取得する関数。	✓			
get_Traffic	モニタ番号	<詳細未定> モニタ結果	当該モニタ番号のモニタ結果を取得する関数。(割当対象に紐付かないモニタ結果取得API)	✓			
get_AllocUnitTraffic	割当対象番号、モニタ番号	<詳細未定> モニタ結果	当該割当対象の当該モニタ結果を取得する関数。		✓		
get_ReportSize	割当対象番号	要求量	当該割当対象の要求量を取得する関数。			✓	
get_ParameterForPolicyMaking	方針決定用パラメータ番号	<詳細未定> 方針決定用パラメータ値	当該方針決定用パラメータ番号の値を取得する関数。(例えば、直近の要求量の総和や、直近の割当量の総和等を想定)				✓
get_AllocUnitParameterForPolicyMaking	割当対象番号、方針決定用パラメータ番号	<詳細未定> 方針決定用パラメータ値	当該割当対象の当該方針決定用パラメータ番号の値を取得する関数。(例えば、当該割当対象の過去の要求量及び割当量の履歴等を想定)				✓
set_GrantSize	割当対象番号、割当量	(なし)	当該割当対象に対する割当量をgrant末尾に追加する関数。(送信開始時刻はgrant処理部が自動計算)	✓	✓	✓	
set_GrantSizeAndStartTime	割当対象番号、割当量、送信開始時刻	(なし)	当該割当対象に対する割当量及び送信開始時刻をgrantに追加する関数。	✓	✓	✓	
set_ParameterForCalc	割当計算用パラメータ番号、<詳細未定> 割当計算用パラメータ値	(なし)	当該割当計算用パラメータ番号の値を設定する関数。(例えば、全割当対象に共通の追加割当量等を想定)				✓
set_AllocUnitParameterForCalc	割当対象番号、割当計算用パラメータ番号、<詳細未定>割当計算用パラメータ値	(なし)	当該割当対象の当該割当計算用パラメータ番号の値を設定する関数。(例えば、当該割当対象への追加割当量等を想定)				✓

5.2.1.1. MFH 向け光モバイル連携 DBA

表 5.2.1.-1 でユースケース 1 の欄にチェックのある API が、MFH 向け光モバイル連携 DBA の利用する API の例である。例えば「get_CooperationConfig」は、連携制御部の動作周期や動作モード、連携先のシステム名、バージョン等を取得する。「get_DataSize」は、帯域割当毎に呼出される。「get_DataSize」は、個別あるいは指定範囲の割当対象をまとめて取得する。

5.2.1.2. MFH 向け NSR-DBA

表 5.2.1.-1 でユースケース 2 の欄にチェックのある API が、MFH 向け NSR-DBA の利用する API の例である。例えば「get_Traffic」「get_AllocUnitTraffic」は帯域割当毎に呼出される。「get_Traffic」は複数 OLT あるいは複数 OSU あるいは複数 CT のフローを、「get_AllocUnitTraffic」は割当対象の全体あるいは割当対象の特定のフローを、それぞれ主に計測対象とする。

5.2.1.3. マス向け SR-DBA (DBA 機能のフルソフト化)

表 5.2.1-1 でユースケース 3a の欄にチェックのある API がマス向け SR-DBA (DBA 機能のフルソフト化) で利用する API の例である。例えば「get_ReportSize」は帯域割当毎に呼出される。「get_ReportSize」は、個別あるいは指定範囲の割当対象の値を取得対象とする。

5.2.1.4. マス向け SR-DBA (DBA 機能の一部ソフト化)

表 5.2.1.-1 でユースケース 3b の欄にチェックのある API がマス向け SR-DBA (DBA 機能の一部ソフト化) で利用する API の例である。API 「get_CalcConfig」の「割当計算部のコンフィグ情報」は、例えば割当対象毎の最大帯域、保証帯域等の設定情報である。「方針決定用パラメータ」は要求量や割当量の履歴等である。

5.3. L2 主信号処理機能

TBD

5.4. 保守運用機能

TBD

5.5. PON マルチキャスト機能

TBD

5.6. 省電力制御機能

TBD

5.7. 周波数/時刻同期機能

TBD

5.8. プロテクション機能

TBD