Technical Specification Group Services and System Aspects **TSGS#23(04)0053**
Meeting #23, Phoenix, USA, 15 - 18 March 2004

Title : **Report of Discussion on Release Structure**

Source: Nortel Networks
(Contact: Iain Sharp)
Agenda: 9.2 - Working Methods

1. Scope and Objective

The 3GPP release process has been a successful vehicle for the delivery of 3GPP releases so far and also for managing legacy GSM releases. However the environment in which 3GPP operates continues to change. New pressures in terms of feature velocity are being felt.

The objective of this document is to present and discuss options for improving the 3GPP release process to better meet the needs of 3GPP members. Presentation of an option in this document does not imply a recommendation to use that option. The "Conclusions" section will be used to present the final findings of the group.

The scope of this document is limited to 3GPP processes linked the production and delivery of specifications. It does not include the organisation of the 3GPP committee structure.

2. Requirements and Constraints

This document discusses ways of changing the 3GPP release structure in order to improve:

- the time taken to add new features to the standard and have them implemented ("velocity")
- the flexibility to choose the feature contents of implementations to allow the market to define which new features are most urgent ("priority")
- the efficiency of the process, particularly in terms of the support cost and the amount of documents that have to be generated and managed

The release structure must allow:

- 3GPP specifications to evolve independently of other bodies (eg OMA) whilst ensuring the support of a mutually agreed set of capabilities across the whole mobile domain.
- the specifications outside 3GPP (e.g. IETF) to evolve independently whilst allowing rapid adoption in 3GPP of the relevant specifications maintained by other bodies without having to tie those specifications to 3GPP releases.
- the constituent parts of the 3GPP system to evolve at a rate that meets their own technical and market requirements

When considering changes to the 3GPP release structure it is also important to maintain:

- the quality of the specifications
- the integrity of the 3GPP system
- good version management/change management
- a simple conceptual framework so that the system can be easily understood

The existing 3GPP release process has proved effective over a long time period, and changes to the process should be approached with caution. Any changes to the existing process also have to be manageable within 3GPP in terms of the effort required to introduce them.

All changes to the 3GPP release process must be reflected in the new versions of the relevant 3GPP procedure description documents such as 3GPP TR 21.801 (Specification drafting rules) and 3GPP TR 21.900 (Technical specification group working methods).

3. Options for Change

Each main subsection of this section presents a different option for improving the 3GPP release process. Some of these options may be mutually compatible, others may be mutually incompatible.

3.1 Early Implementation of Features

"Features may be implemented when standardisation of that feature is completed whether or not the corresponding release is approved"

- stated principle at 3GPP SA plenary

Most vendor companies do not implement all aspects of a release as a block. Instead they pick individual features based on their priorities and those of their customers. Therefore features that are urgently required by operators may be implemented quickly even if they are only included in a late release of the standard.

"Early implementation of features" means to implement an individual feature before the 3GPP release which contains that feature is approved. Therefore features, whenever feasible, should be built as standalone entities, making them independent of other features of the same 3GPP release while maintaining the system integrity and where feasible the system Specification structure. This topic is important because arguments often occur over whether it is necessary to wait for the completion of a 3GPP release before implementing individual features contained in that release.

The position quoted above suggests that it is not in fact necessary to wait for the whole release. In principle individual features can be implemented early. Despite this companies are reluctant to embrace early implementations and therefore 3GPP release dates are still perceived as a gating factor on the availability of new features.

Why are releases considered to be important? A number of reasons can be identified:

- A 3GPP release creates a baseline at which all included specifications are in approved status and considered as complete with all the work items belonging to that release.
- Dependencies and interactions may exist with other features in the release. These may prevent early implementations.
- Protocol version control mechanisms may link features to a new version of the protocol which can only be implemented once the whole release is completed (eg Application-contexts in MAP)
- The processes to stabilise and "freeze" individual features in the standards are unclear. Therefore people making early implementations risk churn in the feature's specifications until the whole release is frozen.
- Commercial terms in contracts that make reference to particular 3GPP releases.

Of the reasons listed above, the protocol version control problem is the most serious. If multiple features are tied together in a new protocol version then it can be technically impossible to make an early implementation of one feature before the other are also ready. The other points are problems but these can be addressed by analysis and risk management in implementing companies.

3.1.1 Dealing with the Protocol Version Problem

The key protocols in GSM/UMTS do allow new features to be introduced in a way that is suitable for early implementation. However several important protocols do include a protocol version control mechanism which can cause problems.

3.1.1.1 Protocol Version Control Problem

Protocols such as MAP sometimes require that extensions are performed by introducing a new version of the protocol. When a new version of a protocol is created this causes particular problems for features that are planned for early implementation and impact the same protocol. This is regardless of whether or not the feature for early implementation is the feature that triggered the decision to generate a new protocol version. The reason for this is explained below.

Most systems that use protocol version control require that a device supporting a particular protocol version is at least able to deal with the full syntax of the new protocol version. Therefore the new protocol version cannot be implemented until its full syntax is stable in the standards.

When a feature is added to 3GPP specifications it is targeted towards a particular release. Therefore the standards documents only specify the signalling for that feature in terms of the protocol versions used in that release. If there are no changes in the protocol versions for any interface impacted by a feature then from a protocol syntax point of view early implementation of the feature should be possible. The feature can be added independently to any node supporting the existing protocol version.

However, when a new protocol version is created then any feature for early implementation that uses the impacted interfaces is constrained by the need to wait for the full syntax of the new protocol version before it can be implemented. The full syntax is not normally certain until all features in the new release are finalised.

In some cases it may not be possible to know in advance whether a new protocol version will be introduced in a particular release. Therefore during the release development what was originally specified on an existing version of a protocol may finally only be standardised for a new protocol version.

3.1.1.2 Consequences for Standardisation

Considering the above discussion we can see that the technical ability to allow early implementation of individual features is not guaranteed. Working groups will be able to take the right design choices if they know in advance that it may be required to make an early implementation of a particular feature. This information should be provided at an early stage of the feature development.

Information from working groups on features' dependencies and linkage to specific protocol versions would be useful to companies assessing the possibility for early implementation. This could be obtained by extending the work item template to include a section for completion by working groups after the feature is completed.

3.1.1.3 Summary on Actions Relating to Protocol Versions

Firstly this discussion has shown that not all features are technically suitable for early implementation. To make sure that features are suitable for early implementation this requirement needs to be identified to working groups.

It is proposed that:

1) Requirements specifications or the work item template should identify features where technical support for early implementation is required or desirable.

2) That the work item template be extended to include a section that can be filled-in after the work item is completed to allow working groups to report on the suitability of the feature for early implementation and its dependencies and interactions.

3.1.2 Freezing of Individual Features

The normal process in 3GPP to stabilise specifications is to agree the "freezing" of a 3GPP release. Changes to frozen releases are only allowed under progressively more restrictive criteria which limit changes to the correction of essential errors.

The applicability of the same idea to individual features prior to the completion of a release is not clear. Though in principle the same process could be followed it is not often done in standards and the tools and processes used do not make specific provision for this event.

Without a good formal process to freeze individual features the early implementation of features will contain an element of risk which must be managed by the implementer and their customers. This risk could be reduced by clarifying and strengthening the formal process for freezing individual work items.

Freezing individual features, or sets of features, must inevitably comprise some means of documentation of those features that are frozen. This documentation should contain at the minimum:

- Date (or 3GPP meeting) after which the agreed work items are frozen
- List of work items that are frozen
- List of impacted specifications
- Indication of whether the whole specification is frozen or are there still other work items that may change the same specification
- Indication of which earlier releases the frozen features are compatible with

The documentation (e.g. a 3GPP TR) of freezing a (set of) features should not define any technical requirements, that is, it does not modify or introduce new stage 1, 2 and 3 documents. The technical requirements for each feature must be the same whether the feature is supported as an early implementation or not.

3.1.3 Summary and Proposal - Early Implementation of Features

As things stand, the early implementation of features involves risks and problems that implementing features from a completed release does not involve. From that point of view it cannot be said that individually completed features outside a release are a sound basis for implementation. As such people will continue to push for complete releases as a basis for implementation. As long as this continues the 3GPP release cycle will be a bottle-neck for new services.

In order to reduce this problem a number of proposals have been made. If these proposals were adopted the problems with early implementations would be reduced. This would alleviate many of the conflicts and pressures that currently apply to feature and release scheduling.

The proposals are to:

- Explicitly plan which features may require an early implementation.
- Take account of protocol properties when designing features for early implementation.
- Capture as soon as possible the possibility for early implementation of individual features.
- Document the “frozen” status of individual features and clarify the processes for the individual freezing of features.

3.2 Creating Multiple Release Streams

Currently 3GPP specifications are released as a single stream. All parts of the system are bundled in to a single release structure. It has been suggested that splitting the

system in to several modules (eg Access Stratum, Non-Access Stratum) and then creating a separate release stream for each module would improve the process.

This section discusses this idea and compares it to the existing arrangement.

3.2.1 Definition and Motivation for Different Release Streams

The motivation for using different release streams is the observation that not all parts of the 3GPP system necessarily develop at the same rate. Trying to synchronise releases across the system pushes some areas faster than they want and holds back others. By creating multiple release streams each module can adjust the release dates to suit their own needs.

An obvious way to structure the different steams would be to separate “RAN” and “Core Network” parts of the system. The IMS which is a kind of overlay on the Packet Domain of the Core Network could also be made in to a separate module. To be strict it must be remembered that 3GPP specifies mobile as well as network aspects. Therefore the “RAN” module is really the “Access Stratum” module and so on. The diagram below illustrates the separation of the system in to three streams.

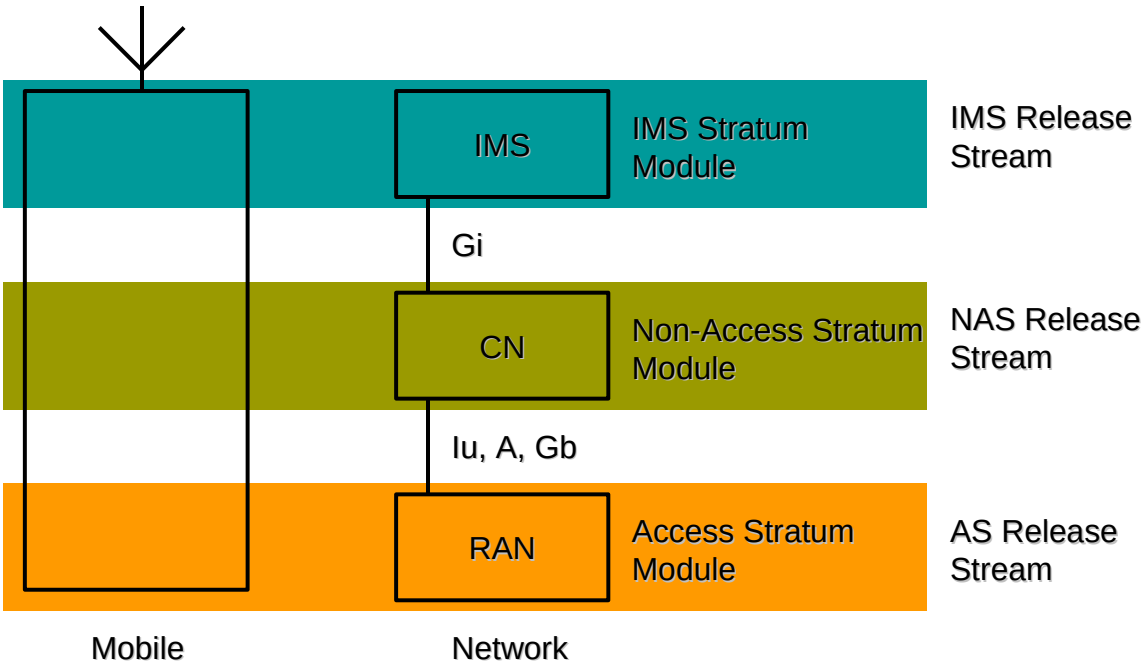

Figure 1 - Separation of Modules and Resulting Release Streams

A release structure based on this split might look as shown below:

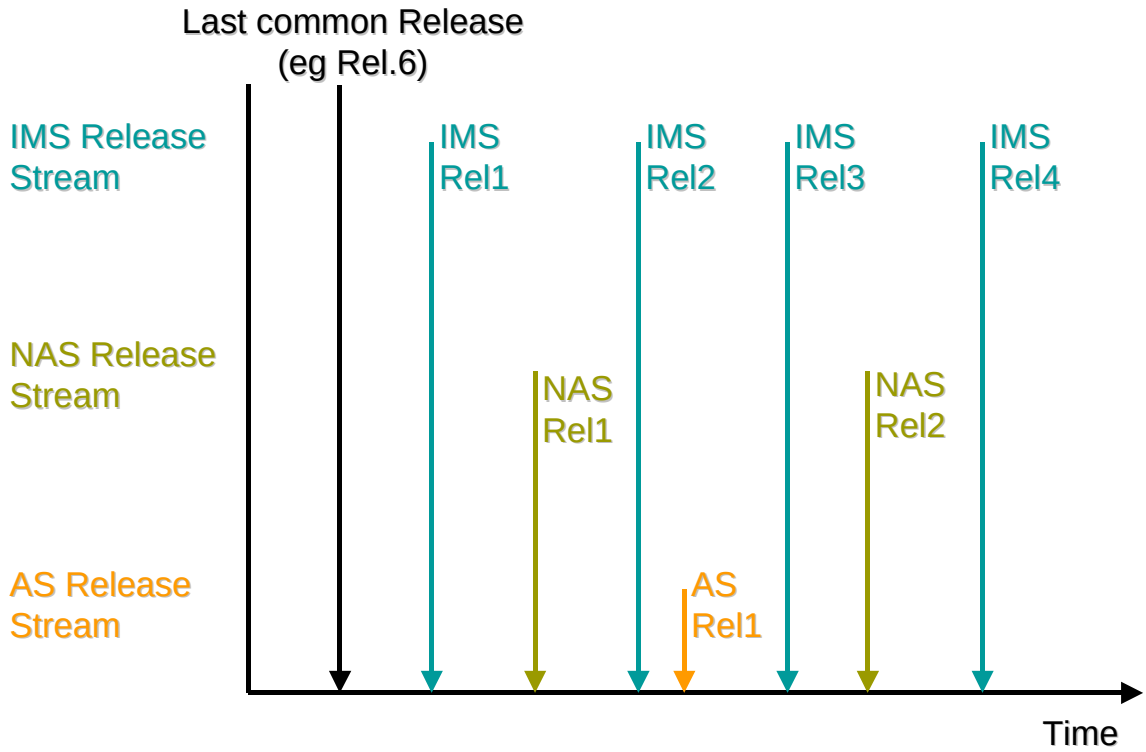

Figure 2 - Example Release Structure with Multiple Streams

These examples greatly simplify the real situation, but a number of conclusions can still be drawn.

3.2.1.1 Cross-Release Compatibility

Releases will need cross-compatibility between different streams.

If you consider figure 2 it can be seen that IMS Rel3 will need to be compatible with NAS Rel1, and that NAS Rel2 must be backwards compatible with IMS Rel3. In addition operational issues may require broader compatibility - eg IMS Rel2 is compatible with NAS Rel2.

Obviously cross-phase compatibility is not a new requirement for GSM/UMTS. However this scheme would extend the existing requirements. Currently cross-phase compatibility is primarily a concern on the radio interface, the SIM-ME interface and inter-PLMN interfaces. Within a PLMN it is normally acceptable to consider that the RAN and the CN are of equivalent functionality. In the multiple stream structure the need to support RANs and CNs of different releases becomes explicit.

3.2.1.2 Inter-Module Interfaces

In order to support the compatibility requirements the interfaces between the modules need to be well defined and the dependencies and requirements well documented. On the infrastructure the inter-module interfaces will probably be chosen to correspond to existing standardised interfaces. However in the mobile station the inter-module

interfaces will probably correspond to internal interfaces for which no explicit documentation exists. Both the ME and the SIM may contain several different modules.

Supporting these inter-module interfaces will require careful work in the standards body. Cooperation between the groups responsible for each module will be needed.

3.2.1.3 Total Number of Specifications Generated

One possible advantage of the multi stream approach is that fewer specifications may be generated if more slowly evolving parts of the system create fewer new releases in their stream.

On average the GSM/UMTS system has had a new release approximately every 18 months in recent history. If we consider a multi stream approach what might be a release rate for each stream? New releases are unlikely to be less than 12 months apart as this is the minimum time needed for significant technical work and to release more often then this would create a great management burden for the stream. Also the releases are unlikely to be more that 24 months apart as this would imply technical work on the module had almost stopped. Therefore it would be expected that even in a multistream model each stream would probably generate a new release at around the same rate as the current system releases. The total number of specifications in use would remain about the same.

3.2.1.4 Complexity

Though the multi-stream release model is reasonably simple it still introduces extra complexity when compared to the current single release stream model. Special handing would be required for the specifications that deal with interfaces between the different modules. The cross-release compatibility of different protocols would have to be documented and managed. Overall the movement to a multi release stream model will add to the management complexity of the system.

3.2.2 Summary and Proposal - Multiple Release Streams

In this analysis the use of multiple release streams would not make any significant reduction in the total number of new specifications being released by 3GPP. Though some simplification would be made in terms of having to coordinate releases over a smaller part of the system this would be replaced by increased complexity in terms of having to deal with inter-module cross-phase issues.

The use of multiple release streams would force the issue of dependencies between different modules within GSM/UMTS to be addressed. Even within the single release stream structure the introduction of tighter management of the dependencies between different modules should be considered to ease deployment issues for new features.

The use of multiple release streams could be simplified by ensuring the regular alignment of releases from individual streams, to provide regular system wide

releases. For example, following the next system wide release (Release 6), the IMS stream could be enhanced independently of the other streams to produce an intermediate sub-release (eg IMS Release 6 bis), and realigned with the rest of the system as part a subsequent system wide release. The development of an intermediate sub-release would have to be constrained to ensure zero impact upon the other streams which form part of the previous system wide release. This approach has the advantage of minimising (eliminating?) inter-stream interaction, whilst enabling the more rapid evolution of individual sub-systems. However the practicalities of coordinating system-wide releases and their frequency needs to be considered.

3.3 Six Monthly Releases

This concept would enable operators to deploy released functionality much more quickly, as we would be able to deliver a release based on whichever features are completed in a 6 month timeframe.

This would rapidly increase the number of releases, and with it the management overhead of producing change requests to all “current” releases. It would also increase significantly the pile of documentation maintained by 3GPP.

4. Conclusions

Section 2 of this report has received careful attention from a number of companies and can be seen as representing a consensus view of the objectives. The level of contribution to section 3 has been variable but some conclusions can be drawn.

4.1 Early Implementation of Features (documented in section 3.1)

This proposes that “Features may be implemented when standardisation of that feature is completed whether or not the corresponding release is approved”. This section describes a mechanism which is already available within the current 3GPP processes where a WI may be frozen independent of the release. However, this mechanism has not yet been invoked or utilized. This mechanism would appear to cover the majority of concerns outlined in this document and meets the requirements and constraints of section 2, with the clarifications to the current allowed procedure documented in section 3.1. Some amendment and enhancement to the current processes documentation are required to fully implement this possibility, e.g. modification of the WI template and/or production of a "release report" etc.

A number of specific proposals to implement varients of the “early implementation of features” concept have been discussed during the generation of the report. If this idea is approved in principle then a further activity would be required to determine the exact method of implementation.

4.2 Multiple Release Streams (documented in section 3.2)

On the whole the transition to multiple release streams does not seem to offer compelling advantages over the current method. It is therefore proposed that the current system-wide release structure remains.

4.3 Six Monthly Releases (documented in section 3.3)

The MCC does not have the resources or industry for that matter, to deal with releases on a fixed 6 monthly basis, e.g the number of mirror CRs, new versions of the specs, release stability, etc. We should not go down the road of having releases determined by calendar date, we have at present a release date based on a reasonable content, which as it turns out, has been on an approximately 12 monthly basis, but the release date is still based on content rather than the timescale.)

Annex A - Revision History (Informative)

Revision	Date	Prepared By	Comments
1	23/12/03	I. Sharp	First Draft
2	8/1/04	I. Sharp	Second draft includes text based on SP-030742 and SP-030782
3	2/2/04	I. Sharp	• Proposed by Nokia email 21/1/04 8.28 (except third change) • Changes proposed by O2 email 28/1/04 12:07 • Deletion of last sentence of old section 2 as proposed by Lucent email 21/9/04 14:19
4	6/2/04	I. Sharp	Corrections to the way previous changes had been incorporated.
5	17/2/04	I. Sharp	• Section 3.4 “Complete Release Independence” deleted as it’s applicability to 3GPP isn’t clear (see my email 5/7/02) • Incorporation of proposals from Motorola email 2/2/04 14:12 • Incorporation of proposals from Nokia (as updated in email 6/2/04 08:45) with some changes to fit better in the existing structure. Text proposed for section 3.5 is incorporated in section 3.1.2. • Incorporation of proposal from O2 email 3/2/04 09:34