Software Configuration Management

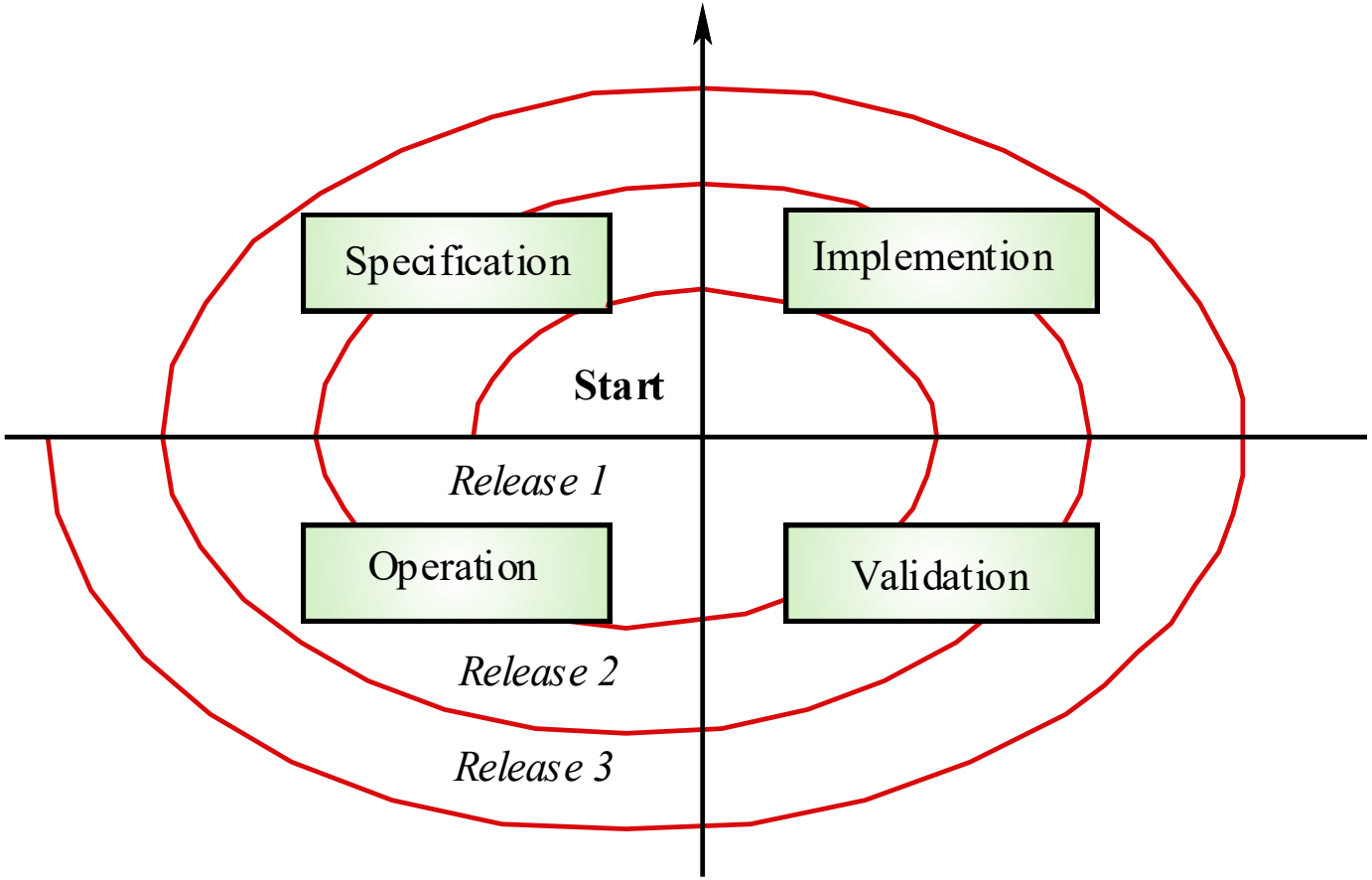
Maintenance is Inevitable

- System requirements are likely to change while the system is being developed because their environment is changing
- Systems are tightly coupled to their environment
- When a system is installed it changes the environment and that can change the system requirements
- The delivered system may not meet its requirements
- Systems must be maintained to remain useful in their environment

Types of Maintenance

- Corrective Maintenance (21%)
  - making changes to repair defects
- Adaptive Maintenance (25%)
  - making changes to adapt software to external environment changes (hardware, business rules, OS, etc.)
- Perfective Maintenance (50%)
  - extending system beyond its original functional requirements
- Preventative Maintenance (4%)
  - modifying work products so that they are more easily corrected, adapted, or enhanced

Spiral Maintenance Model



Maintenance Costs

- Usually greater than the development costs (2 to 10 times as much in some cases)
- Affected by both technical and non-technical factors
- Increase as software is maintained and system corruption is introduced
- Aging software can have high support costs (e.g. old languages, compilers, etc.)

Maintenance Developer Tasks

- Understand system.
- Locate information in documentation.
- Keep system documentation up to date.
- Extend existing functions.
- Add new functions.
- Find sources of errors.
- Correct system errors.
- Answer operations questions.
- Restructure design and code.
- Delete obsolete design and code.
- Manage changes.

Maintenance can be tough

- Limited understanding of hardware and software (maintainer).
- Management priorities (maintenance may be low priority).
- Technical problems.
- Testing difficulties (finding problems).
- Morale problems (maintenance is boring).
- Compromise (decision making problems).

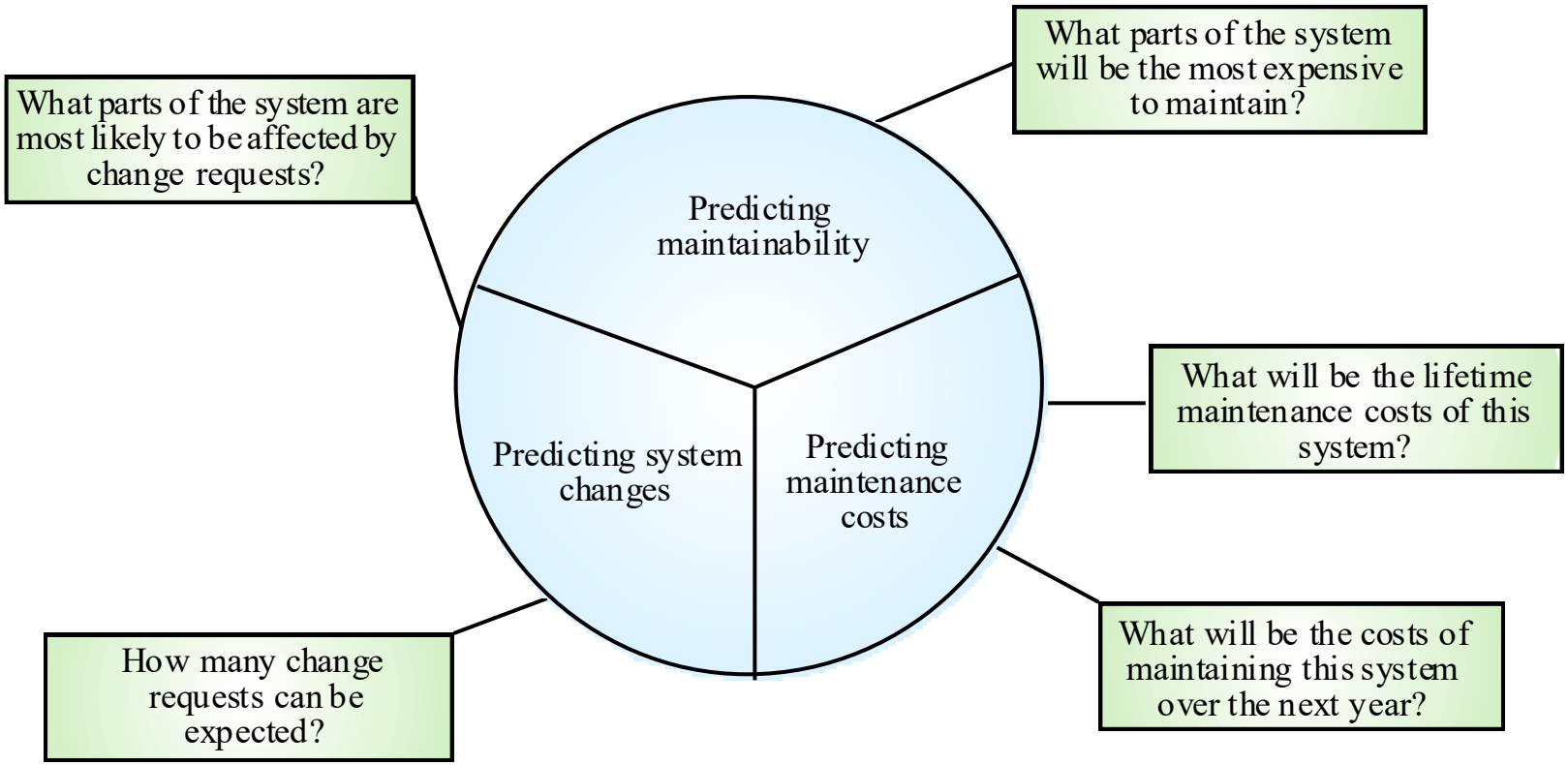
Maintenance Cost Factors

- Staff turnover
  - no turnover usually means lower maintenance costs
- Contractual responsibility
  - developers may have no contractual obligation to maintain the delivered system and no incentive to design for future change
- Staff skills
  - maintenance staff are often inexperienced and have limited domain knowledge
- Program age and structure
  - as programs age their structure deteriorates, they become harder to understand and change

Maintenance Prediction

- Concerned with determining which parts of the system may cause problems and have high maintenance costs
- Change acceptance depends on the maintainability of the components affected by the change
- Implementing changes degrade system and reduces its maintainability
- Maintenance costs depends on number of changes
- Costs of change depend on maintainability

Maintenance Prediction



Maintenance Complexity Metrics

- Predictions of maintainability can be made by assessing component complexities
- Most maintenance efforts only affect a small number of system components
- Maintenance complexity depends on
  - complexity of control structures
  - complexity of data structures
  - module size

Maintenance Tools

- Text editors (better than punch cards).
- File comparison tools.
- Compilers and linkage editors.
- Debugging tools.
- Cross reference generators.
- Complexity calculators.
- Control Libraries.
- Full life cycle CASE tools.

Software Configuration Management

- Software changes are inevitable
- One goal of software engineering is to improve how easy it is to change software
- Configuration management is all about change control.
- Every software engineer has to be concerned with how changes made to work products are tracked and propagated throughout a project.
- To ensure quality is maintained the change process must be audited.

Software Configuration Items

- Computer programs
  - source
  - executable
- Documentation
  - technical
  - user
- Data
  - contained within the program
  - external data (e.g. files and databases)

Baselines

- A work product becomes a baseline only after it is reviewed and approved.
- A baseline is a milestone in software development marked by the delivery of one or more configuration items.
- Once a baseline is established each change request must be evaluated and verified before it is processed.

Sources of Change

- New market conditions dictate changes to product requirements or business rules
- New customer needs demand modification of data, functionality, or services
- Business reorganization causes changes in project priorities or SE team structure
- Budgetary or scheduling constraints require system to be redefined

Change Requests

- Requests can come from users, customers, or management
- Change requests should be carefully analyzed as part of the maintenance process before they are implemented
- Some changes requests must be implemented urgently due to their nature
  - fault repair
  - system environment changes
  - urgently required business changes

Change Prediction

- Predicting the number of changes requires understanding the relationships between a system and its environment
- Tightly coupled systems require changes whenever the environment changes
- Factors influencing the system/environment relationship
  - number and complexity of system interfaces
  - number and volatility of system requirements
  - business processes where the system is uses

Configuration Management Tasks

- Identification
  - tracking changes to multiple SCI versions
- Version control
  - controlling changes before and after customer release
- Change control
  - authority to approve and prioritize changes
- Configuration auditing
  - ensure changes are made properly
- Reporting
  - tell others about changes made

Change Control Process - 1

- Change request is submitted and evaluated to assess its technical merit and impact on the other configuration objects and budget
- Change report containing the results of the evaluation is generated
- Change control authority (CCA) makes the final decision on the status and priority of the change based on the change report

Change Control Process - 2

- Engineering change order (ECO) is generated for each change approved
  - ECO describes the change, lists the constraints, and criteria for review and audit
- Object to be changed is checked-out of the project database subject to access control parameters for the object
- Modified object is subjected to appropriate SQA and testing procedures

Change Control Process - 3

- Modified object is checked-in to the project database and version control mechanisms are used to create the next version of the software
- Synchronization control is used to ensure that parallel changes made by different people don’t overwrite one another

Configuration Management Team

- Analysts.
- Programmers.
- Program Librarian.

Change Control Board

- Customer representatives.
- Some members of the Configuration management team.

Programmer’s View - 1

- Problem is discovered.
- Problem is reported to configuration control board.
- The board discusses the problem
  - is the problem a failure?
  - is it an enhancement?
  - who should pay for it?
- Assign the problem a priority or severity level, and assign staff to fix it.

Programmer’s View - 2

- Programmer or analyst
  - locates the source of the problem
  - determines what is needed to fix it
- Programmer works with the librarian to control the installation of the changes in the operational system and the documentation.
- Programmer files a change report documenting all changes made.

Change Control Issues

- Synchronization (when?)
- Identification (who?)
- Naming (what?)
- Authentication (done correctly?)
- Authorization (who O.K.'d it?)
- Routing (who's informed?)
- Cancellation (who can stop it?)
- Delegation (responsibility issue)
- Valuation (priority issue)

Software Configuration Audit - 1

- Has the change specified by the ECO been made without modifications?
- Has an FTR been conducted to assess technical correctness?
- Was the software process followed and software engineering standards applied?

Software Configuration Audit - 2

- Do the attributes of the configuration object reflect the change?
- Have the SCM standards for recording and reporting the change been followed?
- Were all related SCI's properly updated?

Configuration Status Report

- What happened?
- Who did it?
- When did it happen?
- What else will be affected by the change?

Version Control Terms

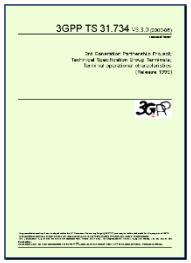
- Entity
  - composed of objects at the same revision level
- Variant
  - a different set of objects at the same revision level and coexists with other variants
- New version
  - defined when major changes have been made to one or more objects

specification as version 0.0.0

Release field

Technical field

Editorial field



The version number

Editorial field x.y.**z**

The Editorial field of the version number is incremented each time an editorial change is made to the document.

It is reset to zero every time the Technical field is updated.

The version number

Technical field x.**y**.z

The Technical field of the version number is incremented each time a technical change is made to the document.

It is reset to zero every time the Release field is updated.

The version number

Release field **x**.y.z

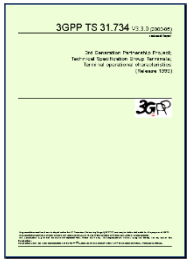
The Release field of the version number is incremented each time major new functionality is made to the system (rather than to the individual document).

The version number

Evolution of the version number of a specification…

v0.0.0
v0.0.1
v0.0.2
v0.1.0
v0.1.1
v0.2.0
v0.3.0
v1.0.0
v1.1.0
v1.2.0
v2.0.0
v3.0.0
v3.1.0
v3.2.0

The drafting process



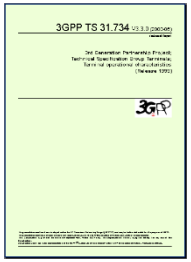
The initial draft is discussed in the working group.

v0.0.0

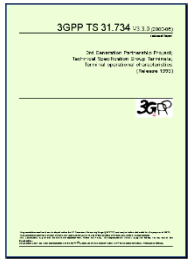
v0.1.0

And a new draft is produced, bearing technical changes.

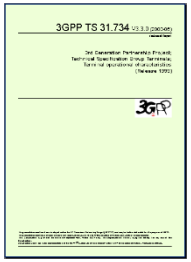
The drafting process



v0.1.0



v0.2.0

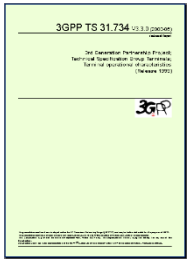


v0.3.0

…

The process is iterative, until …

The drafting process

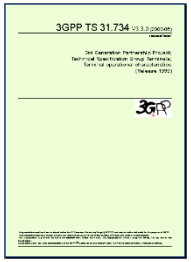


v0.8.0

… the working group is happy with the draft.

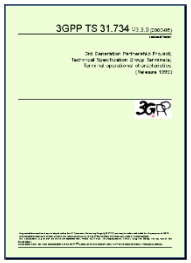
Note that the draft may not be complete, merely acceptable to be aired in a wider forum. As a guideline, the draft should be at least 50% complete to be raised to version 1.0.0.

The drafting process



v0.8.0

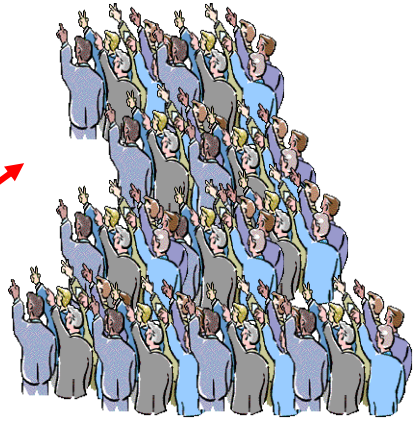
When the draft is “ready”, it is upgraded to version 1.0.0.

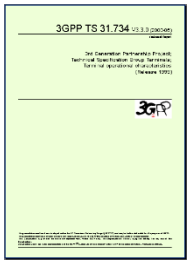


v1.0.0

Note that v1.0.0 is technically identical to the previous 0.y.z document.

The drafting process

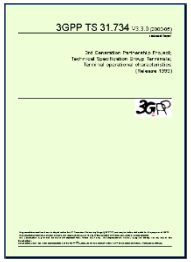




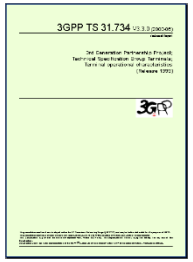
Draft 1.0.0 is presented <u>for information</u> to the plenary TSG (Technical Body).

v1.0.0

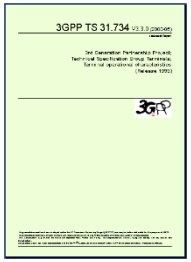
The drafting process



v1.0.0



v1.1.0

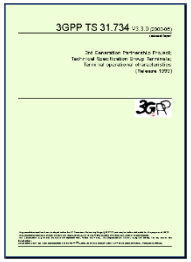


v1.2.0

…

The document returns to the working group, and drafting continues until …

The drafting process

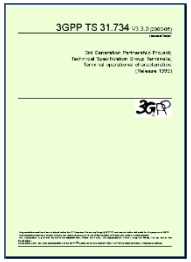


v1.5.0

… the working group believes the draft to be stable enough to come under formal “change control”.

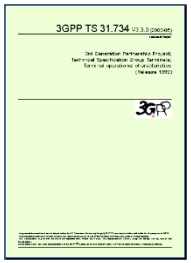
Note that the draft may *still* not be complete, merely ready to come under formal change control. As a guideline, the draft should be at least 80% complete to be raised to version 2.0.0.

The drafting process



v1.5.0

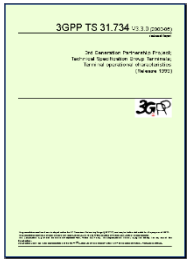
When the draft is “ready”, it is upgraded to version 2.0.0.



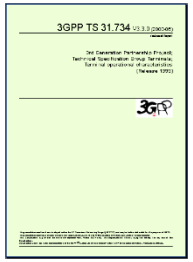
v2.0.0

Note that v2.0.0 is technically identical to the previous 1.y.z document.

The drafting process



v2.0.0



v2.1.0

v2.2.0

…

If the TSG does not approve the draft, it may return to the working group for further refinement. This is exceptional.

Change Control

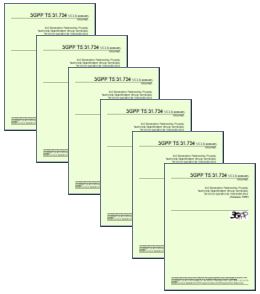
The “system” is composed of a coherent set of related specifications.

For technical and commercial reasons, it may be desirable to divide the standardization process into a number of discrete phases or “Releases”.

Change Control

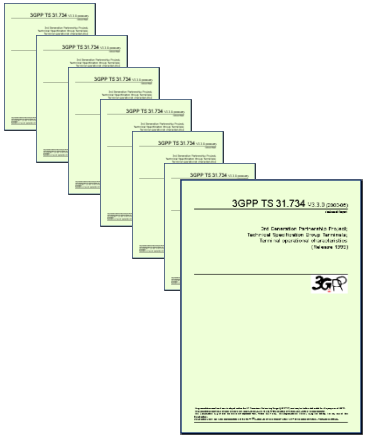
Once a specification has come under change control, the working group and the rapporteur **no longer have the right to update the specification**.

Change Control



But it is still possible to develop the standard further, to add the missing parts, and to correct errors and omissions as the overall system becomes better defined.

Change Control



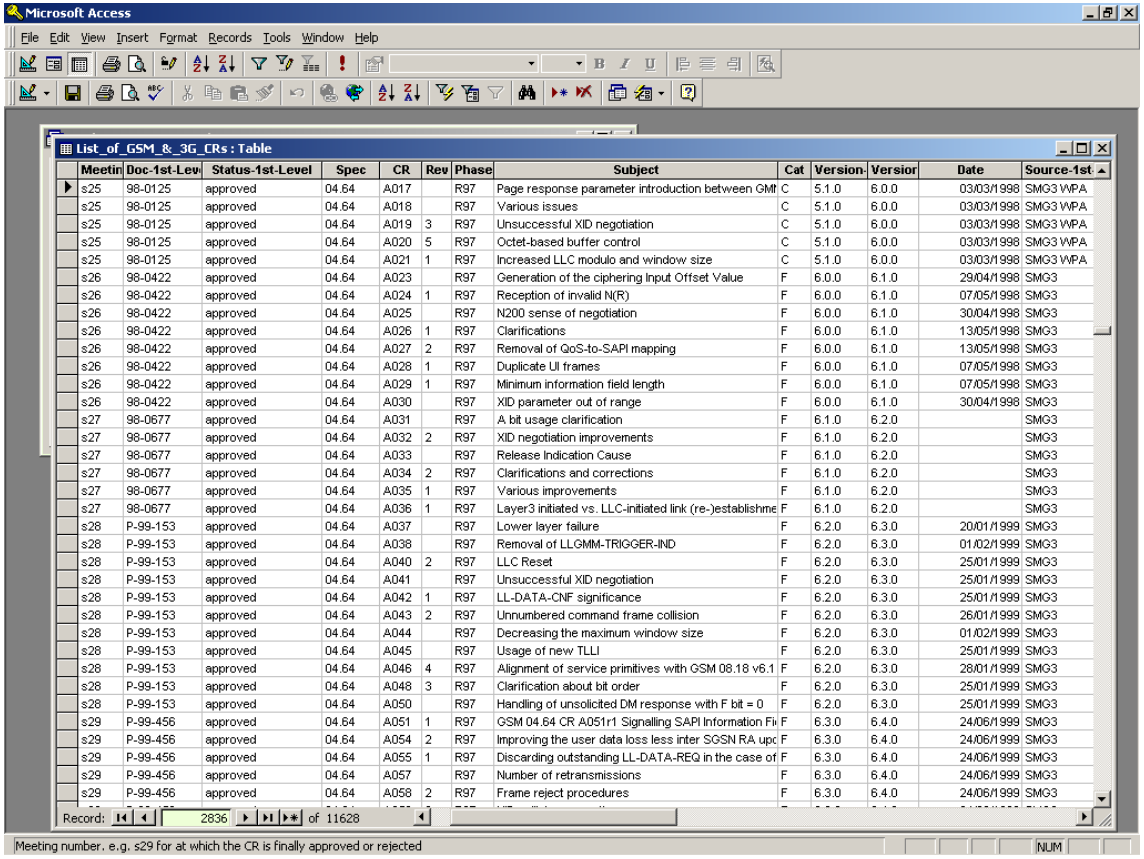
v3.0.0

Consider an individual standard …

If the responsible working group wishes to make a change to it, *however small*, …

Change Control

CRs are registered in a database maintained by the Support Team.



Microsoft Access

File Edit View Insert Format Records Tools Window Help

List_of_GSM_&_3G_CRs : Table

Meetin	Doc-1st-Lev	Status-1st-Level	Spec	CR	Rev	Phase	Subject	Cat	Version-	Versior	Date	Source-1st
s25	98-0125	approved	04.64	A017		R97	Page response parameter introduction between GMI	C	5.1.0	6.0.0	03/03/1998	SMG3 WPA
s25	98-0125	approved	04.64	A018		R97	Various issues	C	5.1.0	6.0.0	03/03/1998	SMG3 WPA
s25	98-0125	approved	04.64	A019	3	R97	Unsuccessful XID negotiation	C	5.1.0	6.0.0	03/03/1998	SMG3 WPA
s25	98-0125	approved	04.64	A020	5	R97	Octet-based buffer control	C	5.1.0	6.0.0	03/03/1998	SMG3 WPA
s25	98-0125	approved	04.64	A021	1	R97	Increased LLC modulo and window size	C	5.1.0	6.0.0	03/03/1998	SMG3 WPA
s26	98-0422	approved	04.64	A023		R97	Generation of the ciphering Input Offset Value	F	6.0.0	6.1.0	29/04/1998	SMG3
s26	98-0422	approved	04.64	A024	1	R97	Reception of invalid N(R)	F	6.0.0	6.1.0	07/05/1998	SMG3
s26	98-0422	approved	04.64	A025		R97	N200 sense of negotiation	F	6.0.0	6.1.0	30/04/1998	SMG3
s26	98-0422	approved	04.64	A026	1	R97	Clarifications	F	6.0.0	6.1.0	13/05/1998	SMG3
s26	98-0422	approved	04.64	A027	2	R97	Removal of QoS-to-SAPI mapping	F	6.0.0	6.1.0	13/05/1998	SMG3
s26	98-0422	approved	04.64	A028	1	R97	Duplicate UI frames	F	6.0.0	6.1.0	07/05/1998	SMG3
s26	98-0422	approved	04.64	A029	1	R97	Minimum information field length	F	6.0.0	6.1.0	07/05/1998	SMG3
s26	98-0422	approved	04.64	A030		R97	XID parameter out of range	F	6.0.0	6.1.0	30/04/1998	SMG3
s27	98-0677	approved	04.64	A031		R97	A bit usage clarification	F	6.1.0	6.2.0		SMG3
s27	98-0677	approved	04.64	A032	2	R97	XID negotiation improvements	F	6.1.0	6.2.0		SMG3
s27	98-0677	approved	04.64	A033		R97	Release Indication Cause	F	6.1.0	6.2.0		SMG3
s27	98-0677	approved	04.64	A034	2	R97	Clarifications and corrections	F	6.1.0	6.2.0		SMG3
s27	98-0677	approved	04.64	A035	1	R97	Various improvements	F	6.1.0	6.2.0		SMG3
s27	98-0677	approved	04.64	A036	1	R97	Layer3 initiated vs. LLC-initiated link (re-)establishme	F	6.1.0	6.2.0		SMG3
s28	P-99-153	approved	04.64	A037		R97	Lower layer failure	F	6.2.0	6.3.0	20/01/1999	SMG3
s28	P-99-153	approved	04.64	A038		R97	Removal of LLGMM-TRIGGER-IND	F	6.2.0	6.3.0	01/02/1999	SMG3
s28	P-99-153	approved	04.64	A040	2	R97	LLC Reset	F	6.2.0	6.3.0	25/01/1999	SMG3
s28	P-99-153	approved	04.64	A041		R97	Unsuccessful XID negotiation	F	6.2.0	6.3.0	25/01/1999	SMG3
s28	P-99-153	approved	04.64	A042	1	R97	LL-DATA-CNF significance	F	6.2.0	6.3.0	25/01/1999	SMG3
s28	P-99-153	approved	04.64	A043	2	R97	Unnumbered command frame collision	F	6.2.0	6.3.0	26/01/1999	SMG3
s28	P-99-153	approved	04.64	A044		R97	Decreasing the maximum window size	F	6.2.0	6.3.0	01/02/1999	SMG3
s28	P-99-153	approved	04.64	A045		R97	Usage of new TLLI	F	6.2.0	6.3.0	25/01/1999	SMG3
s28	P-99-153	approved	04.64	A046	4	R97	Alignment of service primitives with GSM 08.18 v6.1	F	6.2.0	6.3.0	28/01/1999	SMG3
s28	P-99-153	approved	04.64	A048	3	R97	Clarification about bit order	F	6.2.0	6.3.0	25/01/1999	SMG3
s28	P-99-153	approved	04.64	A050		R97	Handling of unsolicited DM response with F bit = 0	F	6.2.0	6.3.0	25/01/1999	SMG3
s29	P-99-456	approved	04.64	A051	1	R97	GSM 04.64 CR A051r1 Signalling SAPI Information Fi	F	6.3.0	6.4.0	24/06/1999	SMG3
s29	P-99-456	approved	04.64	A054	2	R97	Improving the user data loss less inter SGSN RA upc	F	6.3.0	6.4.0	24/06/1999	SMG3
s29	P-99-456	approved	04.64	A055	1	R97	Discarding outstanding LL-DATA-REQ in the case of	F	6.3.0	6.4.0	24/06/1999	SMG3
s29	P-99-456	approved	04.64	A057		R97	Number of retransmissions	F	6.3.0	6.4.0	24/06/1999	SMG3
s29	P-99-456	approved	04.64	A058	2	R97	Frame reject procedures	F	6.3.0	6.4.0	24/06/1999	SMG3

Record: 2836 of 11628

Meeting number, e.g. s29 for at which the CR is finally approved or rejected

NUM