



# Back to the Basics

Prof. Andreas Prinz  
Høgskolen i Agder



# Background

- SDL is a world-leader
  - the most popular graphical representation
- But SDL has become big & complex
  - Implementation of tools is expensive
  - New users need to be skilled software engineers
- Features missing for testing and for PDU encoding



# SDL Task Force Goals

- Identify the simplest, useful SDL-subset
  - Derived from first principles
  - Compatible with SDL
  - Cheaper for tool suppliers to support
  - Easier for new users to learn
- Add the simplest, useful enhancements for
  - Test Specifications
  - PDU Encoding
- Ensure automated graphical presentation
- Promote SDL



# The SDL Task Force

- Editorial board
  - William Skelton, Vangelis Kollias, Alkis Yiannakoulis, Qing Li, Andreas Prinz
- Contributing members
  - Review and comment on work of editorial board
- Recognized by the SDL-Forum
- Developed the language SDL+



# SDL+: The simplified SDL!

- The SDL Task Force designed SDL+ to be
  - accessible to users with typical requirements
  - bring implementation and testing closer together (tool integration)
- The starting point is: *What is a state machine?*
  - And everything for implementation follows...
- Then to ask: *What is a test?*
  - And everything for testing follows...



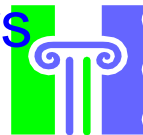
# So What is a State Machine?

- The most basic mechanism in SDL+
- Interface: Gates
- State-Event Matrix
  - Defines behaviour for the state machine
  - States: general & start & default
  - Event constraints
    - gate constraint and
    - PDU constraint
    - wildcards



# Extensions

- Static Dimensioning
  - Arrays of states machines
  - Arrays of gates
  - index variables for gates
- PDU encoding
  - conversion from and to ASN.1
  - e.g. ENCODE v TO pdu USING decode\_rule
- New features for testing
  - Test Suite, Test Group, Test Case, Test Purpose, Matching Mechanism, Verdict, Test Suite Parameters



# SAVE

- long discussion in SDL Task Force
- Use-cases not convincing so far
- Required for:
  - packet buffering
  - State reduction for signals that can arrive in arbitrary order
  - Out of Context Signals
  - Procedures with states
- More operators needed for queue handling
  - clear, push, pop, FIFO





# Current State

- Foundation: July 2003
- Version 1.1: August 2003
- Version 1.2 (internal)
- Version 1.3: March 2004
  - **SDL+** is lightweight ('simplest, useful') and suitable for heavyweight applications
  - It is almost fully compatible with an SDL-subset
  - Improved scoping & data-hiding
  - **SDL+** has features for testing state machines!



# Future

- Update with SAM'04 comments planned 3Q2004
- Graphical representation planned 1Q2005
- Use **SDL+** to promote SDL
- Increase acceptance by users
  - Getting started, tutorials, examples
- Improve the documentation
- Cooperate with the ITU-T!

