Simulation of Systems Development Processes – Why, What and How?

LUCAS Breakfast Presentation – 20 April 2011

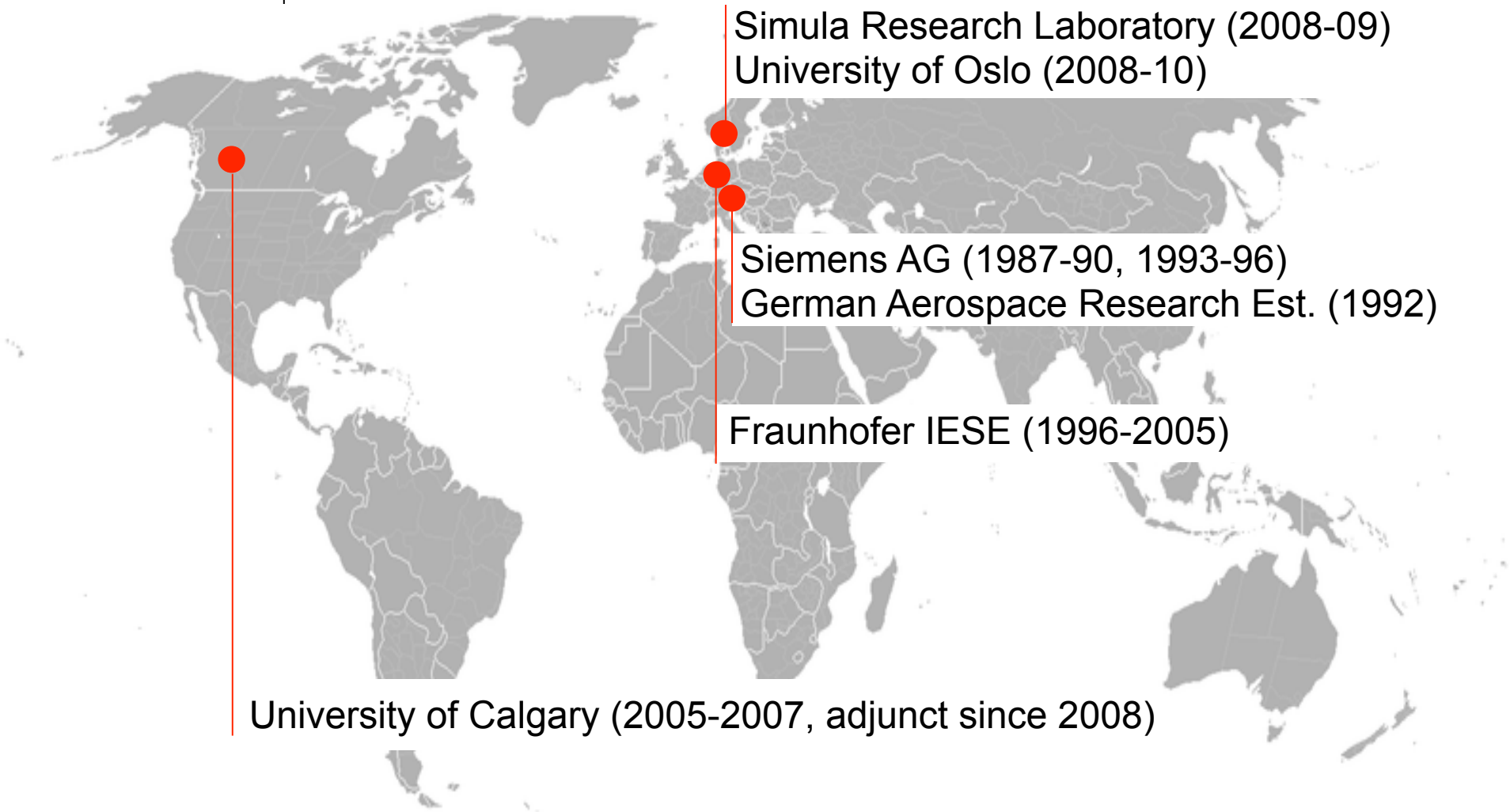
Dietmar Pfahl

Content

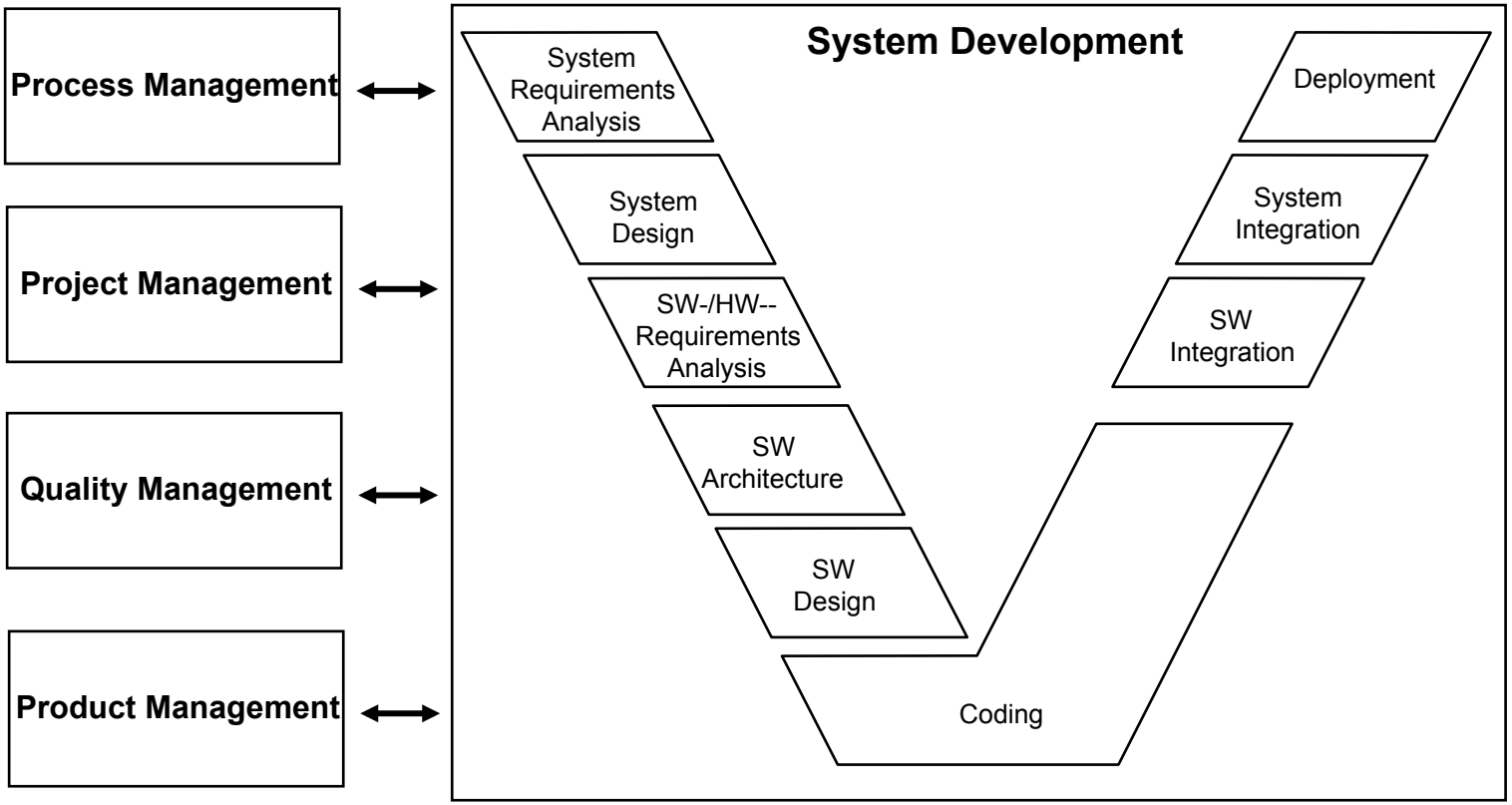
1. **Background**
2. **Current Research**
   - SPS* – Why?
   - SPS – What?
   - SPS – How?
3. **Future Research Goals**
   - VSEL**

*SPS = Software Process Simulation
**VSEL = Virtual Software Engineering Laboratory

My Background – Industry and Academia

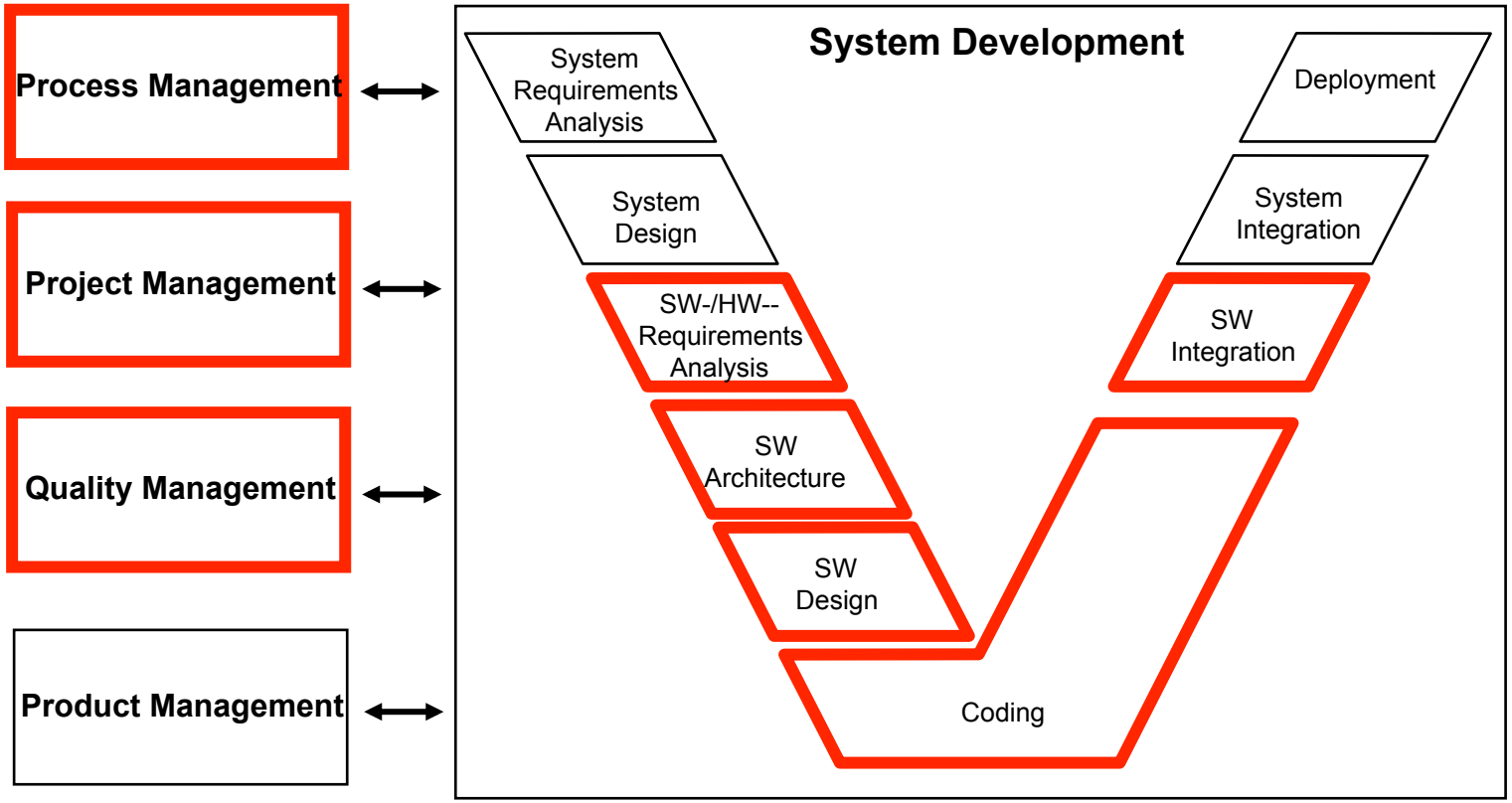


My Background – Applied Research



My Background – Applied Research

Software process simulation (Siemens, DaimlerChrysler, Bosch, Chartwell, 1994-95, 1997-98, 2004, 2005, 2008-10)



Content

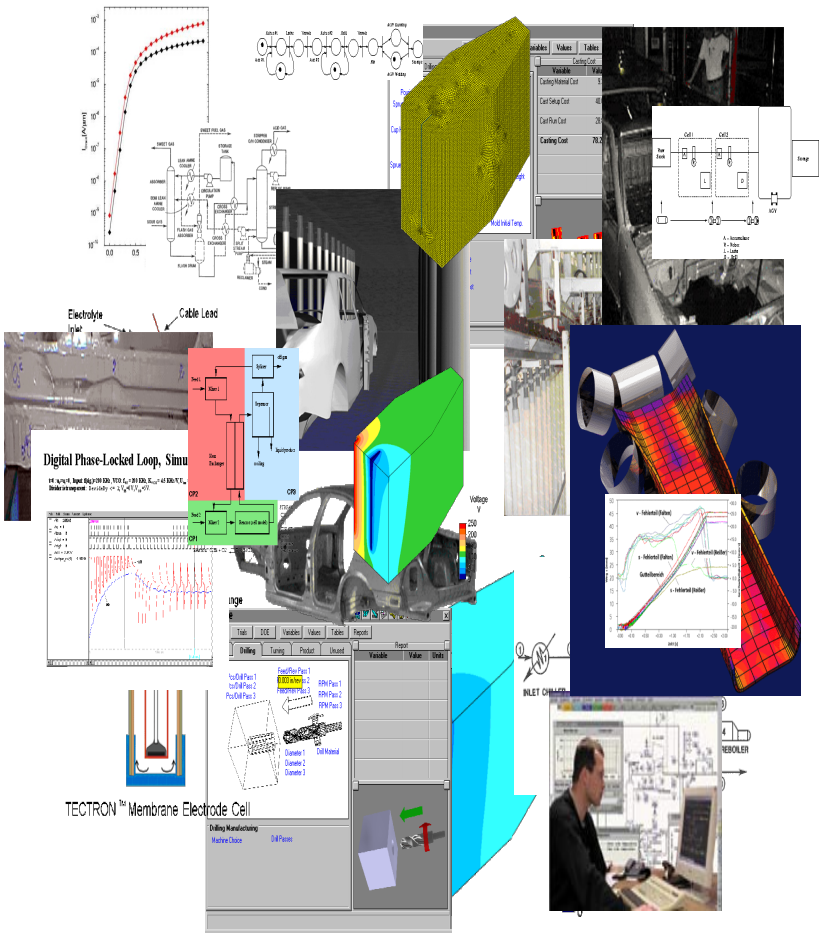
1. **Background**
2. **Current Research**
   - SPS* – Why?
   - SPS – What?
   - SPS – How?
3. **Future Research Goals**
   - VSEL**

*SPS = Software Process Simulation
**VSEL = Virtual Software Engineering Laboratory

SPS Models – Why?

It's standard in other engineering disciplines:

- **Industrial building processes**
- **Product engineering processes**
- **Chemical engineering processes**
- **Pharmaceutical engineering processes**
- **Hydrological engineering processes**
- **Manufacturing processes**
- **Biological/Ecological processes**
- **Flight simulators**
- **Aeronautics**
- **….**



SPS Models – Why?

- **To <u>understand</u>**
  - …the interesting characteristics of an existing or desired (complex, socio-technical) development system (process, project)
  - SPS models as analysis tools (e.g., for risk analysis)
- **To <u>predict</u>**
  - …the interesting characteristics of the system (process, project) by analysing its model's performance
  - SPS models as planning tools
- **To <u>evaluate</u>**
  - …the properties of alternative systems (process, project) by comparing its models' performances
  - SPS models as planning tools
- **To <u>communicate</u>**
  - …the design and functioning of the system (process, project) to others
  - SPS models as documentation and education tools

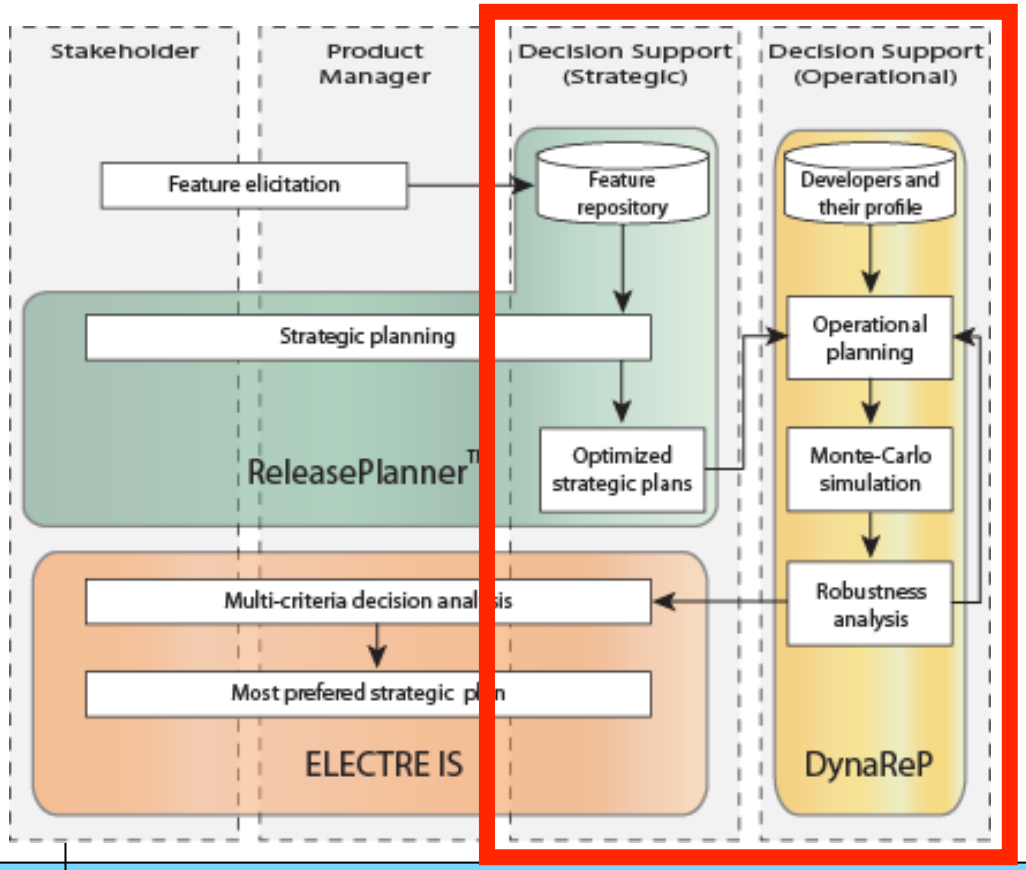
Content

1. **Background**
2. **Current Research**
   - SPS* – Why?
   - SPS – What?
   - SPS – How?
3. **Future Research Goals**
   - VSEL**

*SPS = Software Process Simulation
**VSEL = Virtual Software Engineering Laboratory

SPS Models – What?

- **Operational Release Planning (ORP)**
  - Example of using SPS for planning, risk analysis, and decision-making
  - Complements existing optimization models for Strategic Release Planning (→ $DECIDE_{Release}$)

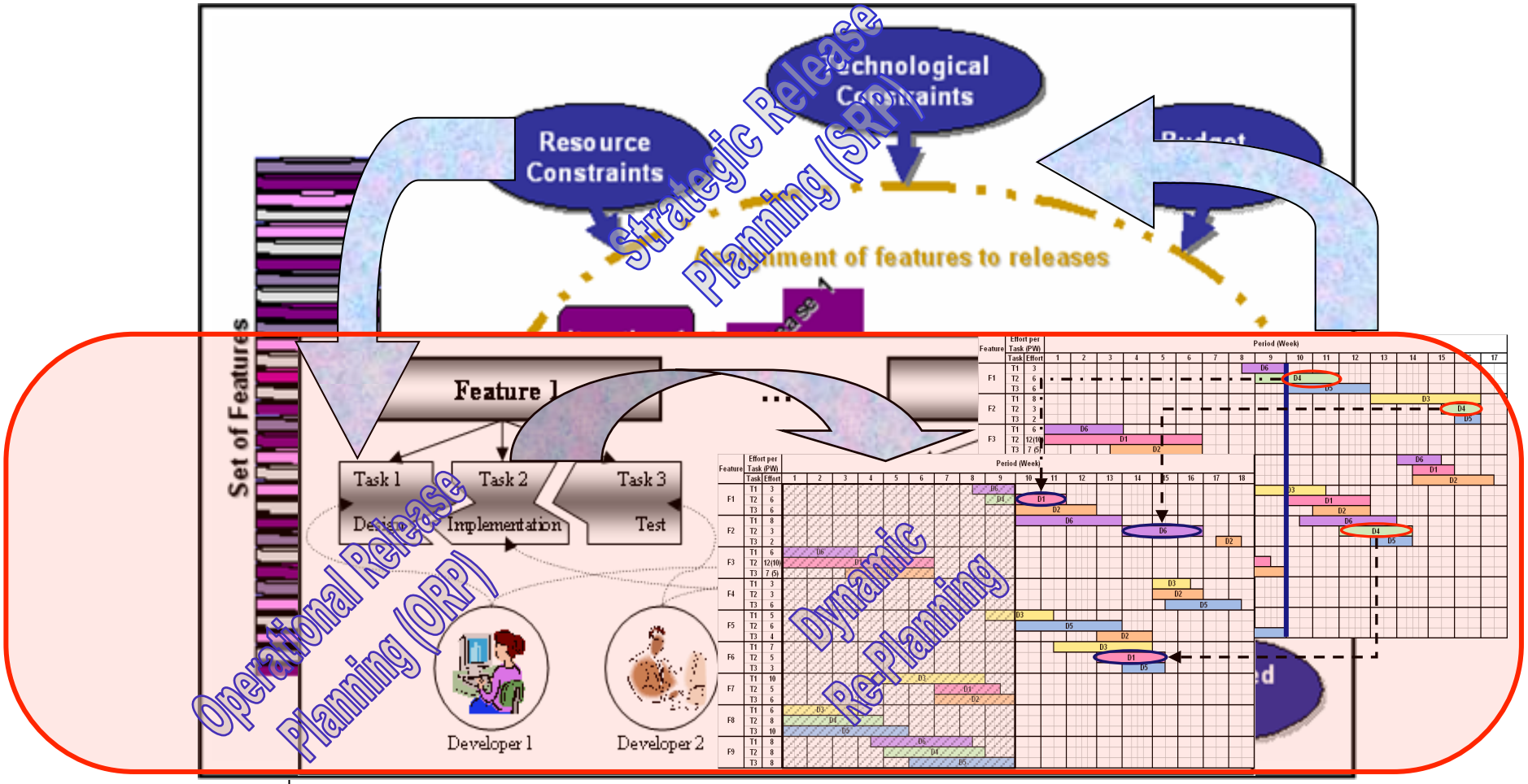


Reasons for ORP re-planning:
- Drop in/out of developers
- Addition/deletion of features
- Adjustment of effort estimates
- Adjustment of productivity estimates
- Adjustment of task dependencies

All the above …
… in any combination
… at any point in time during release (repeatedly)

SPS Models – What?

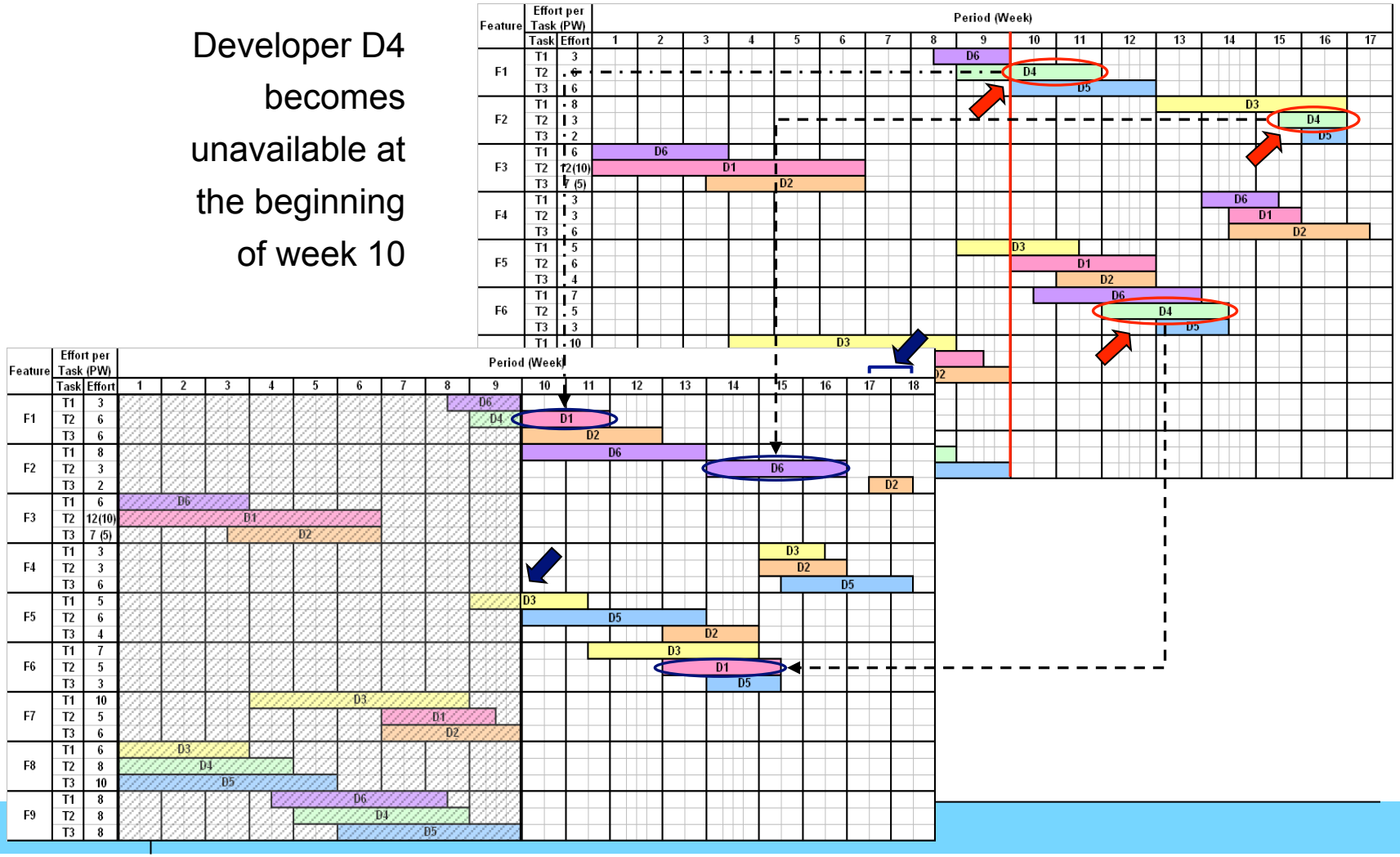
- ORP: Planning and Re-Planning in the context of SRP



SPS Models – What?

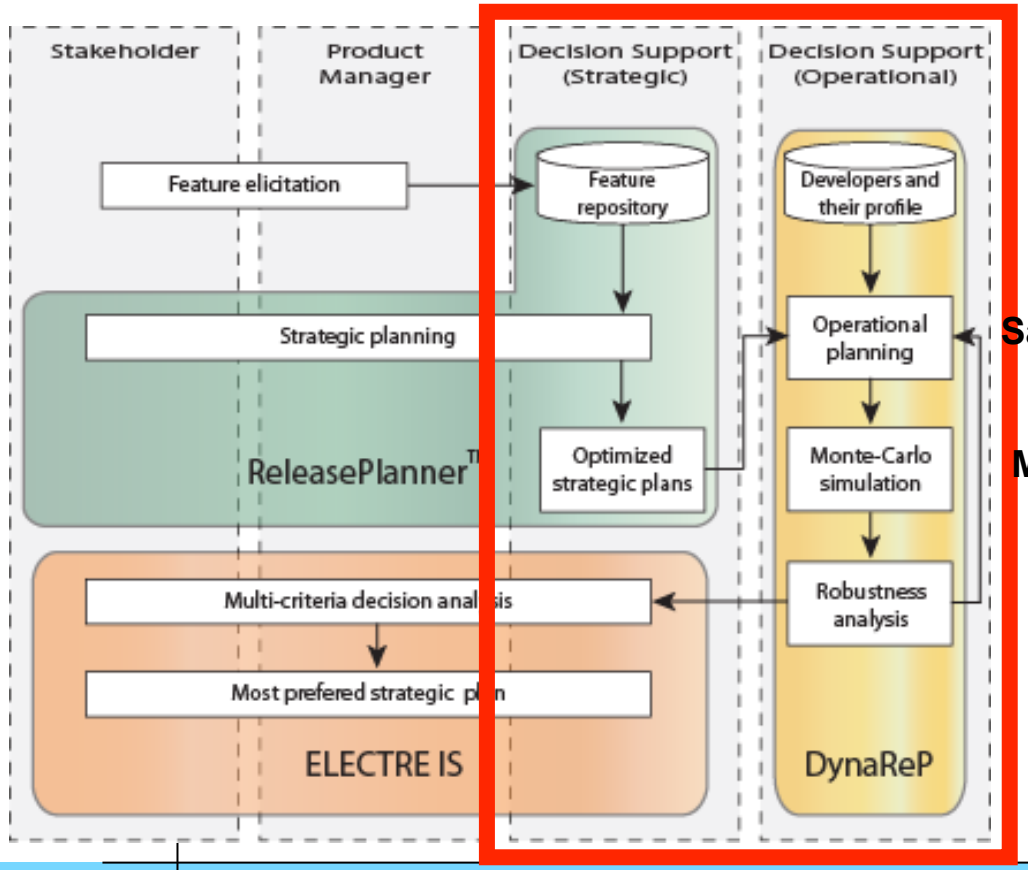
- **Re-Planning Example (enlarged):**

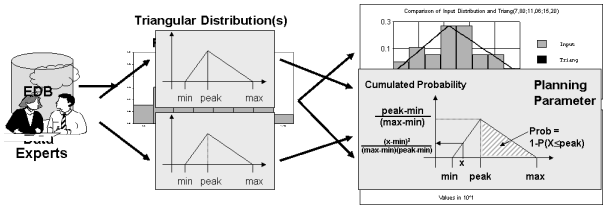
Developer D4 becomes unavailable at the beginning of week 10



SPS Models – What?

- **Operational Release Planning**
  - Example of using SPS for planning, risk analysis, and decision-making
  - Complements existing optimization models for Strategic Release Planning ($DECIDE_{Release}$)

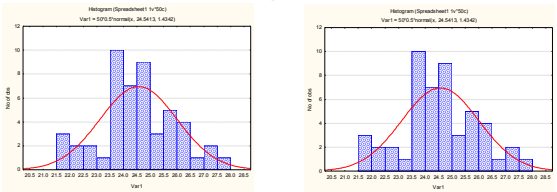




Sampling of input parameter values from probability distributions

Monte-Carlo & Process Simulation

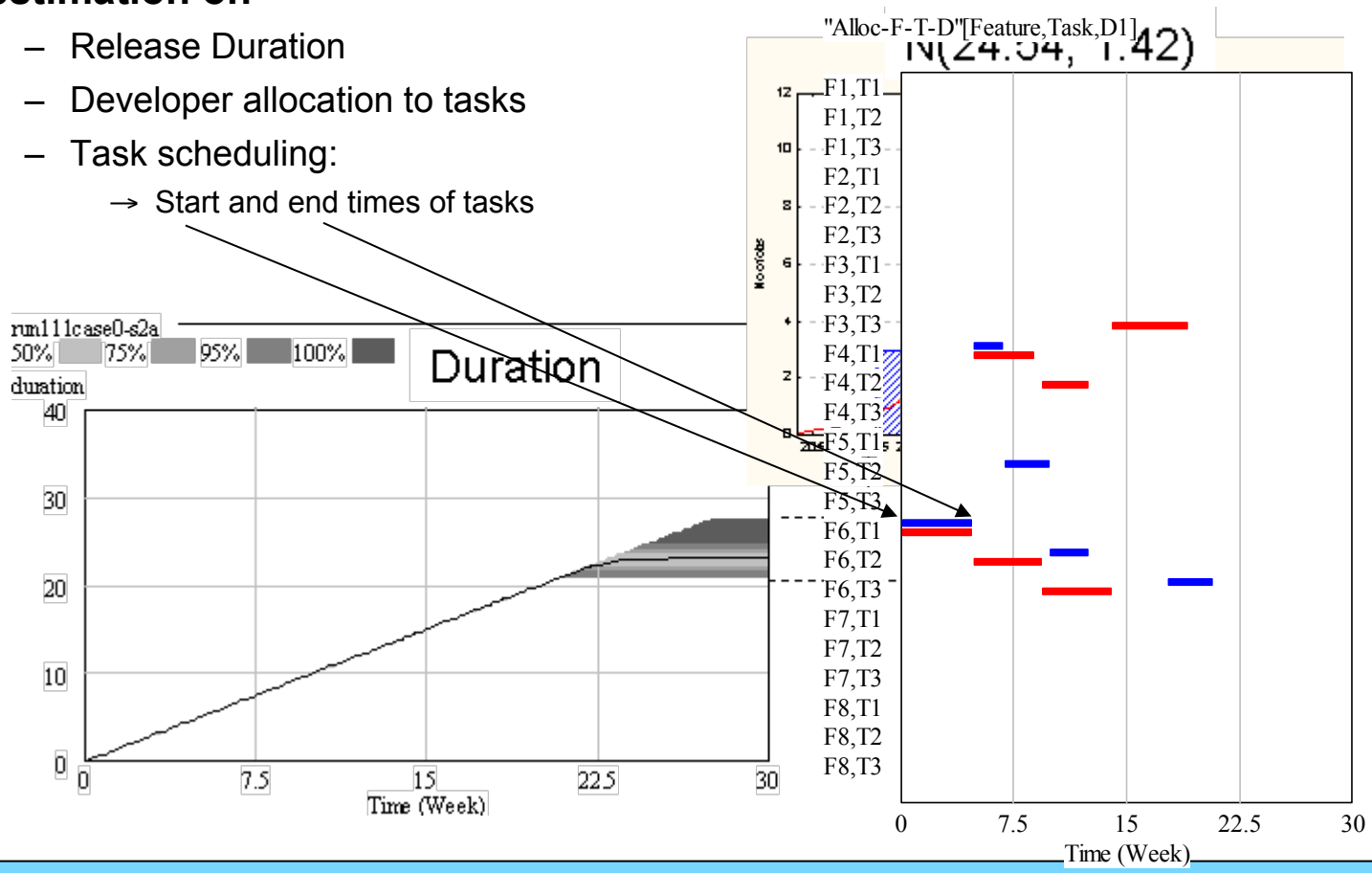
Distributions of output parameter values



SPS Models – What?

- **Risk Analysis for Release Planning**

  To analyze impact of task effort and developer productivity over/under-estimation on
  - Release Duration
  - Developer allocation to tasks
  - Task scheduling:
    - → Start and end times of tasks



SPS Models – What?

- **Baseline Case**

Table 1a. Feature/Task-specific efforts

		eff(i, j) [person-week]							
		f(1)	f(2)	f(3)	f(4)	f(5)	f(6)	f(7)	f(8
Task	t(1)	3	8	6	3	5	7	10	6
	t(2)	6	3	10	3	6	5	5	8
	t(3)	6	2	5	6	4	3	6	10

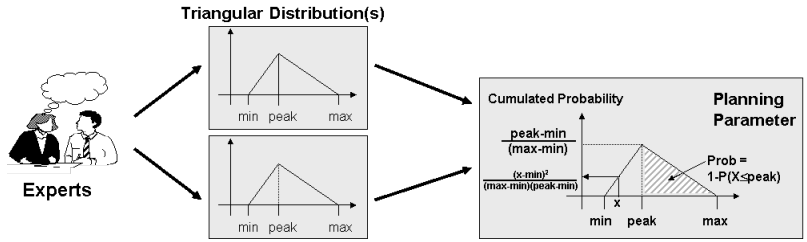
Table 1b. Developer-specific productivities

		p(k, j) [no unit]					
		d(1)	d(2)	d(3)	d(4)	d(5)	d(6)
Task	t(1)	1.5	1	2	0	0.5	2
	t(2)	2	1.5	1	2	1.5	1
	t(3)	1	2	0	1.5	2	1

- **Triangular Distributions**

Table 2. ProSim/ORP risk factor variation

Case	Variation Parameter	Min	Max
Case 1	Effort	-10%	20%
Case 2	Effort	-20%	20%
Case 3	Effort	-20%	40%
Case 4	Effort	-40%	40%
Case 5	Productivity	-20%	10%
Case 6	Productivity	-20%	20%
Case 7	Productivity	-40%	20%
Case 8	Productivity	-40%	40%
Case 9	Mixed	Case 1 + Case 5	
Case 10	Mixed	Case 2 + Case 6	
Case 11	Mixed	Case 3 + Case 7	
Case 12	Mixed	Case 4 + Case 8	



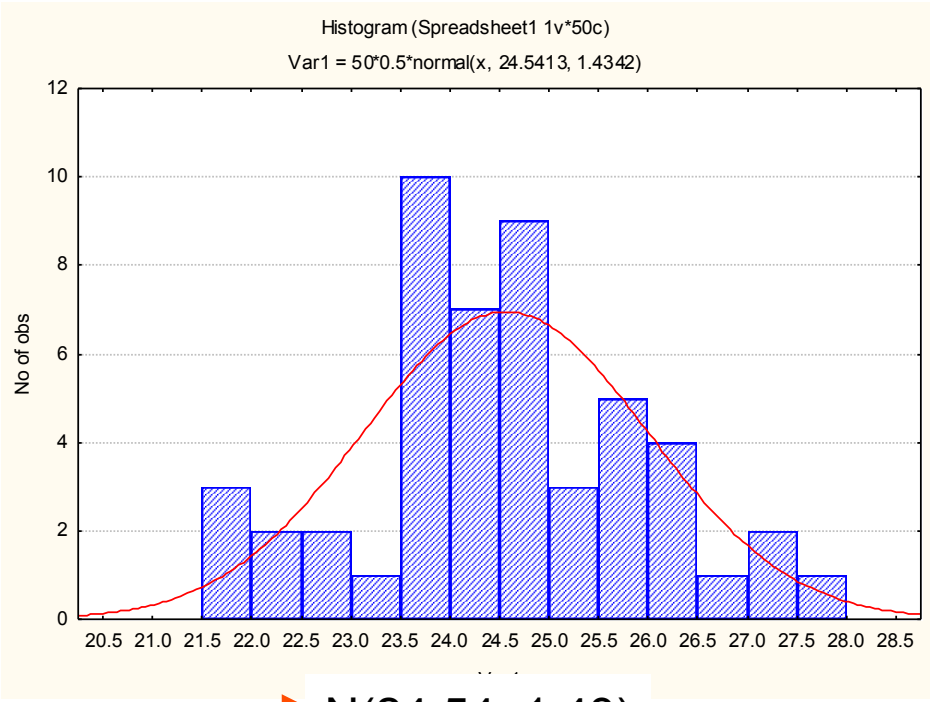
SPS Models – What?

- **Effect on Duration**

Table 3a. Summary statistics of ORP performance parameter Dur_run

Case	Baseline Duration [weeks]	Run-specific duration Dur_run [weeks]	
		Mean	Standard dev.
Case 1	22.71875	23.811	1.611
Case 2	22.71875	23.146	1.608
Case 3	22.71875	24.755	2.210
Case 4	22.71875	23.526	2.428
Case 5	22.71875	23.780	1.012
Case 6	22.71875	23.075	1.250
Case 7	22.71875	24.849	2.034
Case 8	22.71875	23.508	2.141
Case 9	22.71875	24.541	1.434
Case 10	22.71875	23.038	2.080
Case 11	22.71875	26.367	2.495
Case 12	22.71875	23.950	3.331

- **Case 9: Distribution of Durations**



N(24.54, 1.43)

SPS Models – What?

- **Effect on Duration**

Case (50 runs each)	Deviation of Effort Mean from Baseline Effort	Deviation of Productivity Mean from Baseline Productivity	Deviation of Duration Mean from Baseline Duration
Case 1	3.4%	0%	4.8%
Case 2	0%	0%	1.9%
Case 3	6.8%	0%	9.0%
Case 4	0%	0%	3.6%
Case 5	0%	-3.3%	4.7%
Case 6	0%	0%	1.6%
Case 7	0%	-6.7%	9.4%
Case 8	0%	0%	3.5%
Case 9	3.4%	-3.7%	8.0%
Case 10	0%	0%	1.4%
Case 11	6.9%	-7.5%	16.1%
Case 12	0%	-0.1%	5.4%

- **Single Sample T-Test**

Table 5. Results from the single sample t-tests comparing the distribution of the average durations in each case with the duration in the baseline case

Case (n=50)	Baseline Duration	Duration Mean	Duration Std. Dev.	p-Value
Case 1	22.71875	23.811	1.611	0.000
Case 2	22.71875	23.146	1.608	0.066
Case 3	22.71875	24.755	2.210	0.000
Case 4	22.71875	23.526	2.428	0.023
Case 5	22.71875	23.780	1.012	0.000
Case 6	22.71875	23.075	1.250	0.049
Case 7	22.71875	24.849	2.034	0.000
Case 8	22.71875	23.508	2.141	0.012
Case 9	22.71875	24.541	1.434	0.000
Case 10	22.71875	23.038	2.080	0.284
Case 11	22.71875	26.367	2.495	0.000
Case 12	22.71875	23.950	3.331	0.012

?

alpha = 0.05

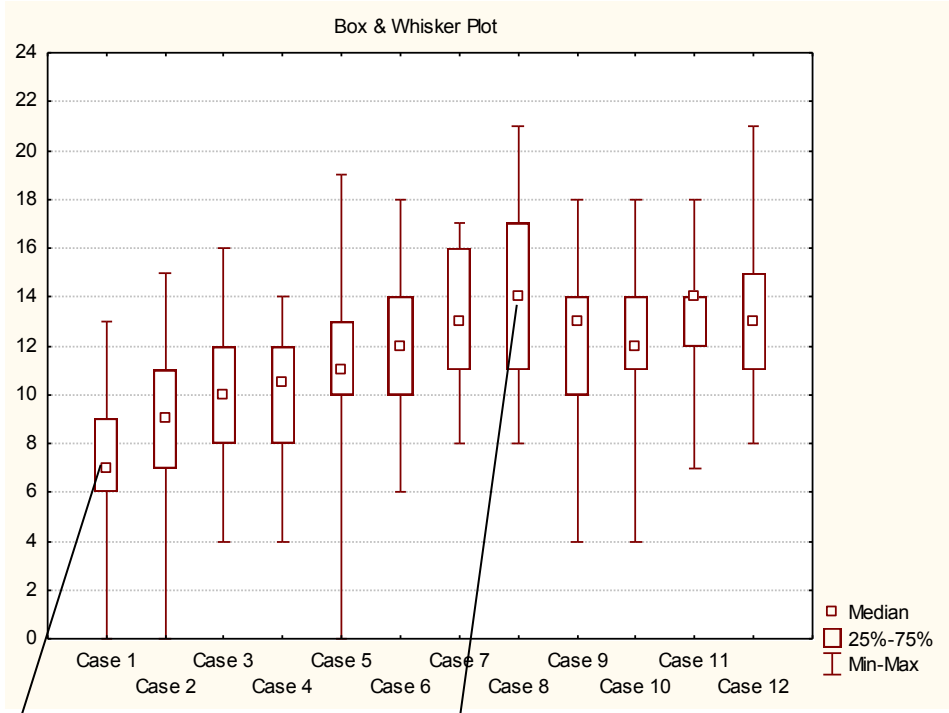
SPS Models – What?

- **Effect on Developer Allocation**

Table 3b. Summary statistics of ORP performance parameter Alloc_diff

Case	Difference in developer allocation Alloc_diff [no unit]	
	Mean	Standard dev.
Case 1	7.2	3.289
Case 2	9.04	3.194
Case 3	9.765	2.688
Case 4	10.14	2.298
Case 5	11.38	3.562
Case 6	12.26	2.776
Case 7	13.3	2.644
Case 8	14.06	3.191
Case 9	11.96	3.201
Case 10	12.16	2.78
Case 11	13.08	2.806
Case 12	13.3	2.652

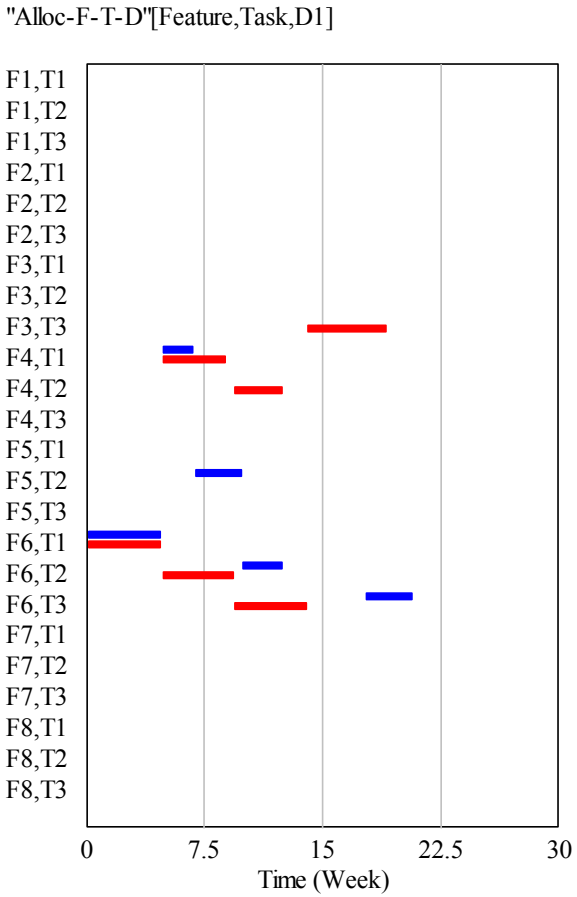
- **Total number of allocations: 8 x 3 = 24**
- **With 6 developers:**



Median: 29% change Median: 58% change

SPS Models – What?

- **Effect on Task Scheduling**



- **Data:** **Dv_diff ∈ [0, 1]**

Table 3c. Summary statistics of ORP performance parameters ST_diff, ET_diff, Dv_diff

Case	ST_diff		ET_diff		Dv_diff	
	Mean	Std. Dev.	Mean	Std. Dev.	Mean	Std. Dev.
Case 1	19.26	11.34	25.44	11.9	0.4214	0.1503
Case 2	27.67	16.67	35.28	17.79	0.5026	0.1533
Case 3	32.72	17.02	42.93	17.76	0.5345	0.136
Case 4	40.76	17.8	52.43	19.52	0.6002	0.1299
Case 5	28.15	17.8	33.71	17.84	0.4958	0.1465
Case 6	34.28	17.55	40.8	17.48	0.5536	0.1344
Case 7	46.01	20.96	54.09	21.32	0.6096	0.1306
Case 8	47.95	20.49	57.06	21.55	0.6368	0.112
Case 9	38.29	18.62	47.41	19.19	0.5742	0.123
Case 10	41.05	16.03	49.71	16.26	0.61	0.0987
Case 11	60.34	22.13	75.17	26.19	0.6902	0.0817
Case 12	62.53	32.56	75.81	30.18	0.7052	0.0717

Content

1. **Background**
2. **Current Research**
   - SPS* – Why?
   - SPS – What?
   - SPS – How?
3. **Future Research Goals**
   - VSEL**

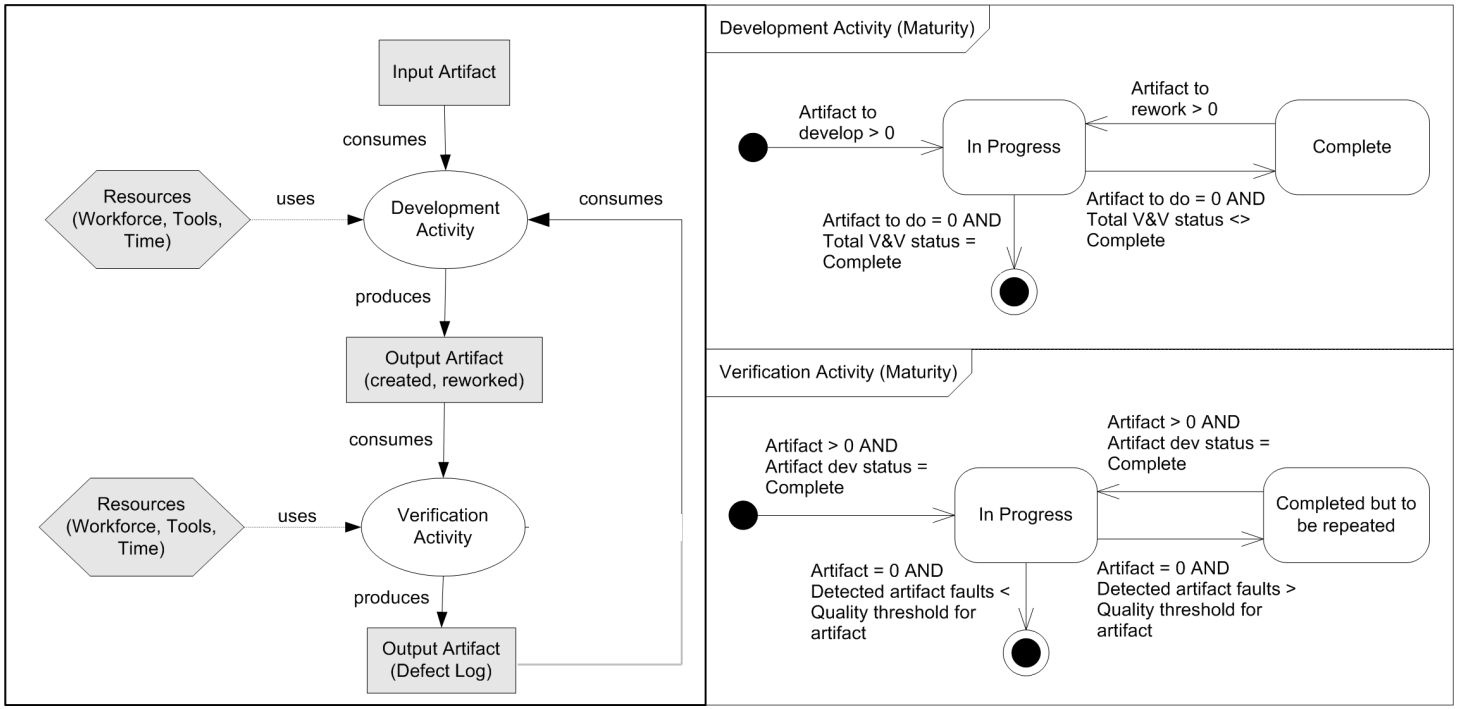
*SPS = Software Process Simulation
**VSEL = Virtual Software Engineering Laboratory

SPS Models – How?

- **A Virtual Software Engineering Laboratory (VSEL)**
  - Exploring the idea of SPS building blocks (macro-patterns)
  - Currently development of an open source MATLAB/SIMULINK library under way (→ Th. Birkhoelzer, Konstanz)
  - Example: Composition of SPS representing the V-model
    - → E.g., used to investigate the effects of various factors (e.g., quantity and quality of personnel resources, effectiveness of inspection and test activities) on project performance (effort consumption, duration, defect density)

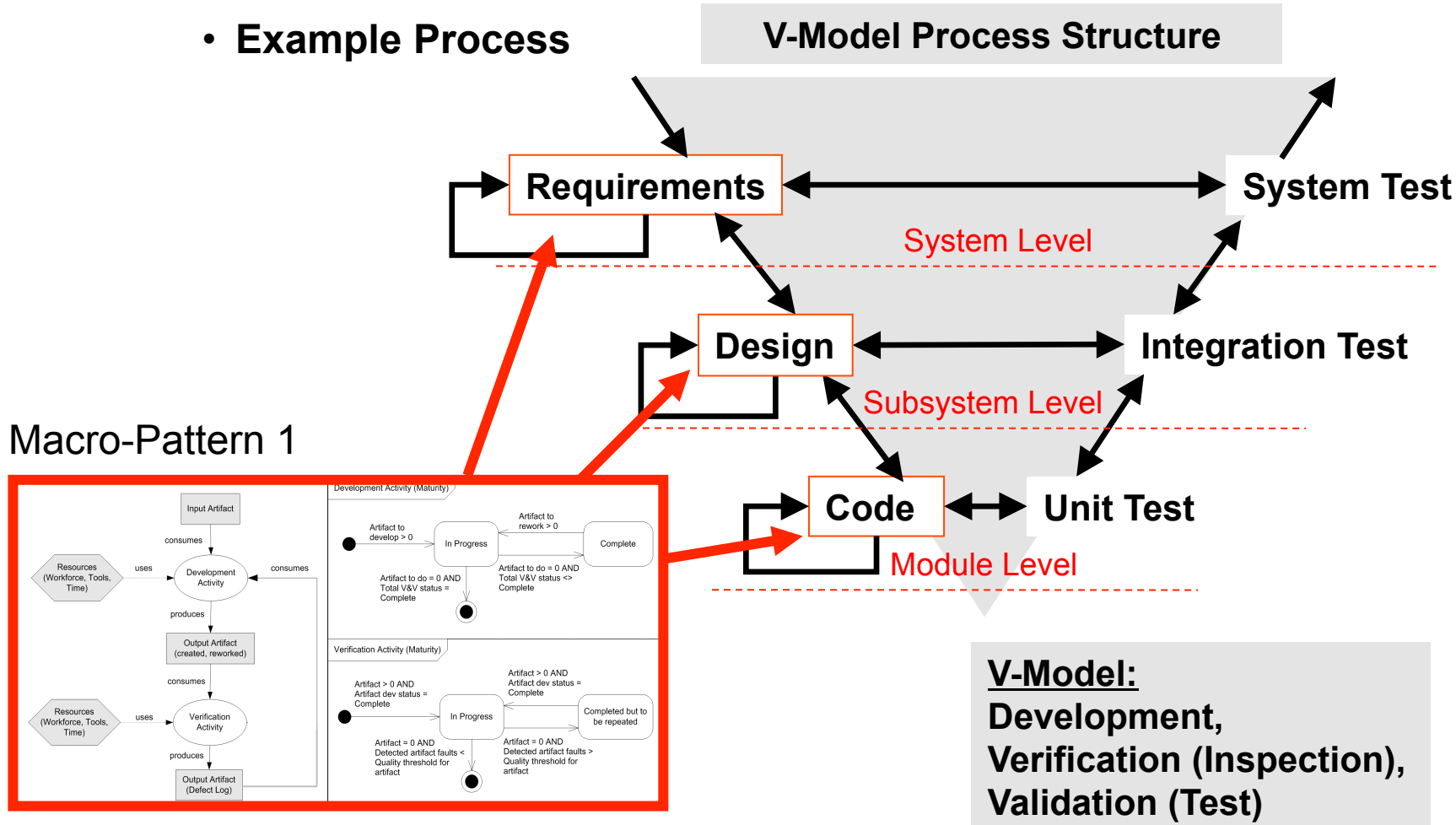
SPS Models – How?

- **Macro-Pattern 1: Development and Verification**



SPS Models – How?

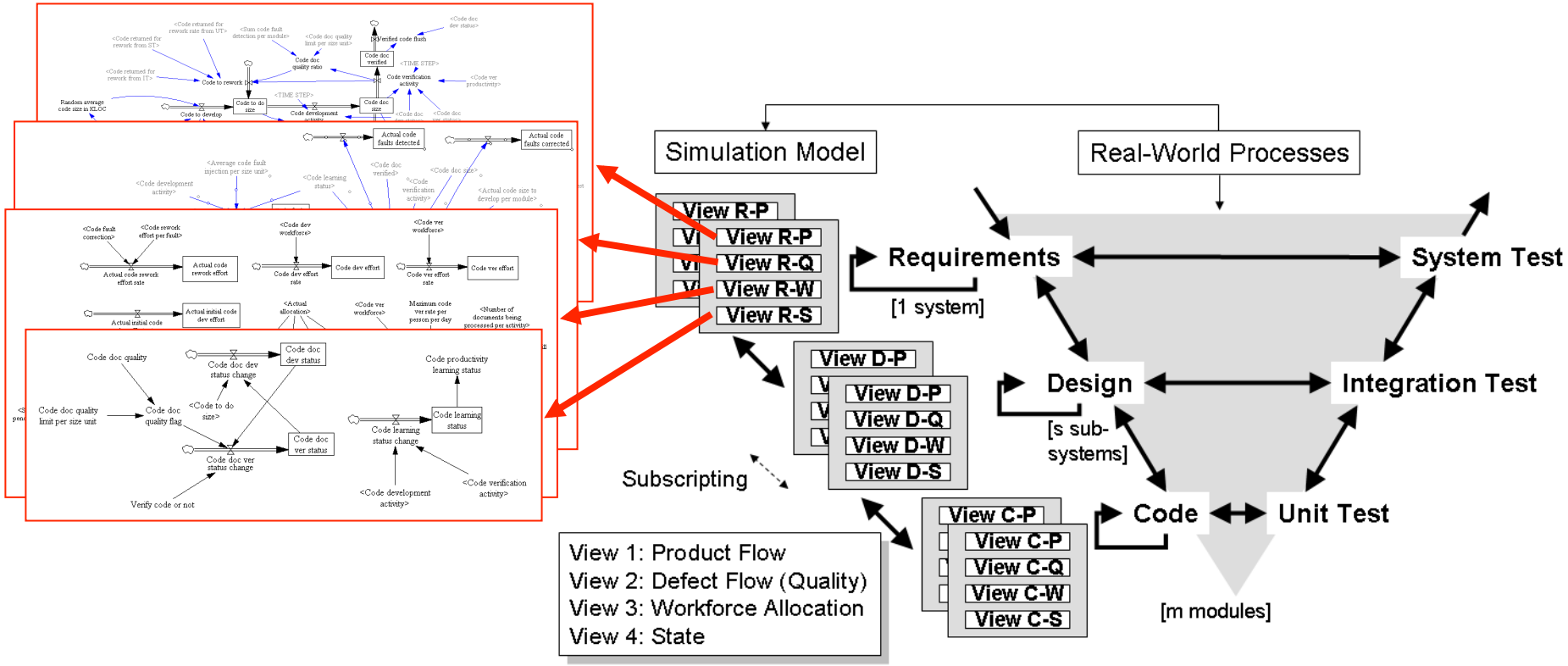
- **Example Process**



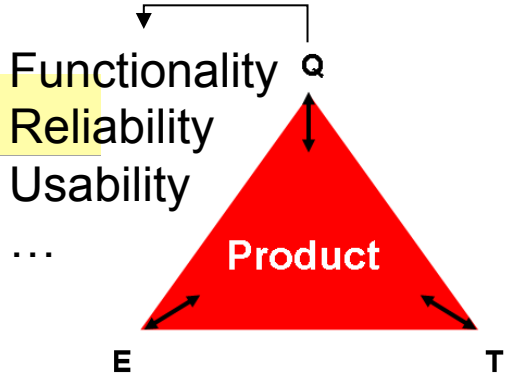
V-Model:
Development,
Verification (Inspection),
Validation (Test)

SPS Models – How?

- **Implementation (System Dynamics, MATLAB/SIMULINK)**



SPS Models – How?



<Skills>

<Learning>

<Policies (Rules)>

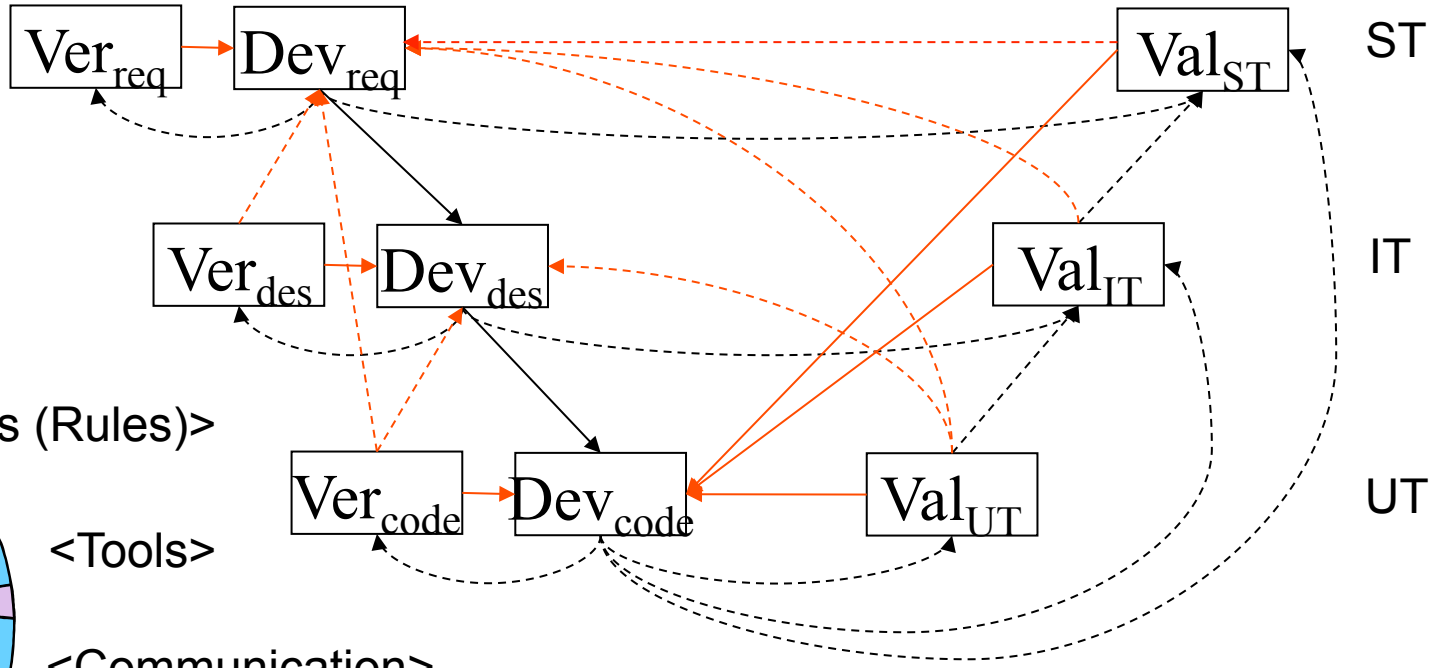
<Tools>

<Communication>

<Techniques>

- **GENSIM 2.0 – Generic Simulator**

  Requirements Analysis and Specification
  Design
  Implementation



SPS Models – How?

Possible Analyses (list not complete):

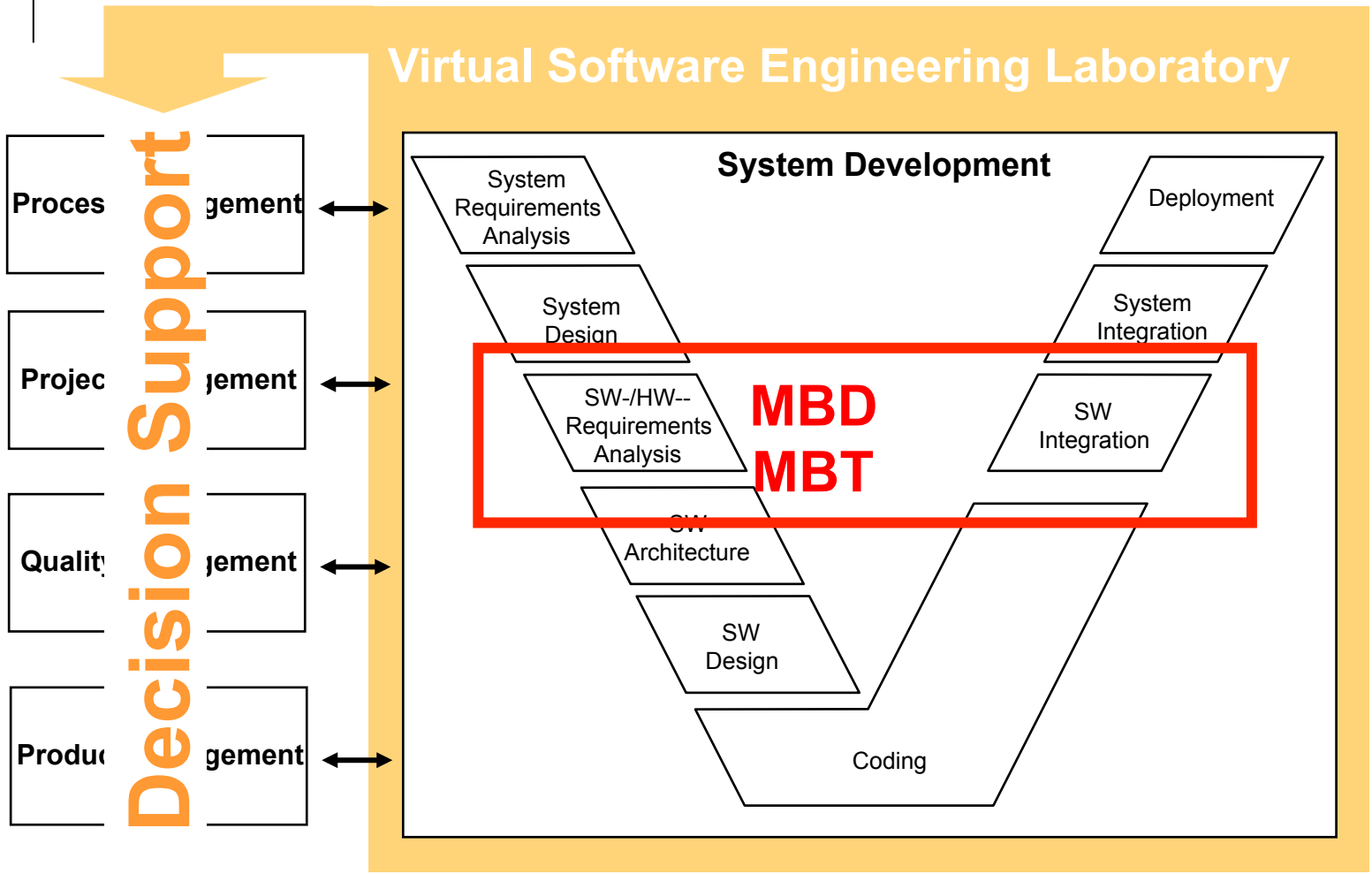
- What combinations (and intensity levels) of development, verification and validation techniques should be applied in a given context to achieve defined time, quality or cost goals?
- How do workforce size and skill variations impact project performance (project duration, effort consumption, code quality)?
- Does investment in training pay off for specific development contexts and goals?
- Do investments in improving development, verification, and validation techniques pay off for specific development contexts and goals?
- What are the promising areas of research for improving development, verification, and validation techniques?

Content

1. **Background**
2. **Current Research**
   - SPS* – Why?
   - SPS – What?
   - SPS – How?
3. **Future Research Goals**
   - VSEL**

*SPS = Software Process Simulation
**VSEL = Virtual Software Engineering Laboratory

Future Research Goals – Big Picture



VSPL – Why/What/How?

Virtual Software Engineering Laboratory (VSEL)

- **Why**
  - To provide a test-bed for evaluating issues or new ideas in a variety of contexts
  - To support decision-making
- **What**
  - To analyse/visualise cost-effectiveness of new development approaches, e.g., model-based development (i.e., model-based testing)
  - To analyse effects of local improvements (e.g., better skills for a specific tasks, better effectiveness of a specific inspection or test technique) or local problems (e.g., unclear or incomplete requirements, requirements volatility) on overall project/process/organisational performance
- **How**
  - Provision of a library of customizable model components macro-patterns)
  - Calibration of customised process simulator based on empirical data and expert knowledge

Tack så mycket!

Questions?

Publication Summary (since 1994)

- **Books/Proceedings + Book Chapters: 8 + 8**
- **Journals: 17**
  - Software Process Improvement & Practice: 5
  - Information and Software Technology: 3
  - The Journal of Systems and Software: 2
  - IJSEKE: 2
  - Empirical Software Engineering: 1
  - IEEE Transactions on Reliability: 1
  - Others: 3
- **Conference and Workshop Proceedings: 51**
  - SEKE (incl. SEDECS): 6
  - ProSim (ICSE-Workshop): 6
  - PROFES: 5
  - ICSP: 4
  - ISESE: 3
  - METRICS: 3
  - ESCOM: 3
  - LSO: 3
  - Others: 18