



# *Construction Process Modelling*

Elisa Marengo ([marengo@inf.unibz.it](mailto:marengo@inf.unibz.it))

Faculty of Computer Science, Free University of Bolzano

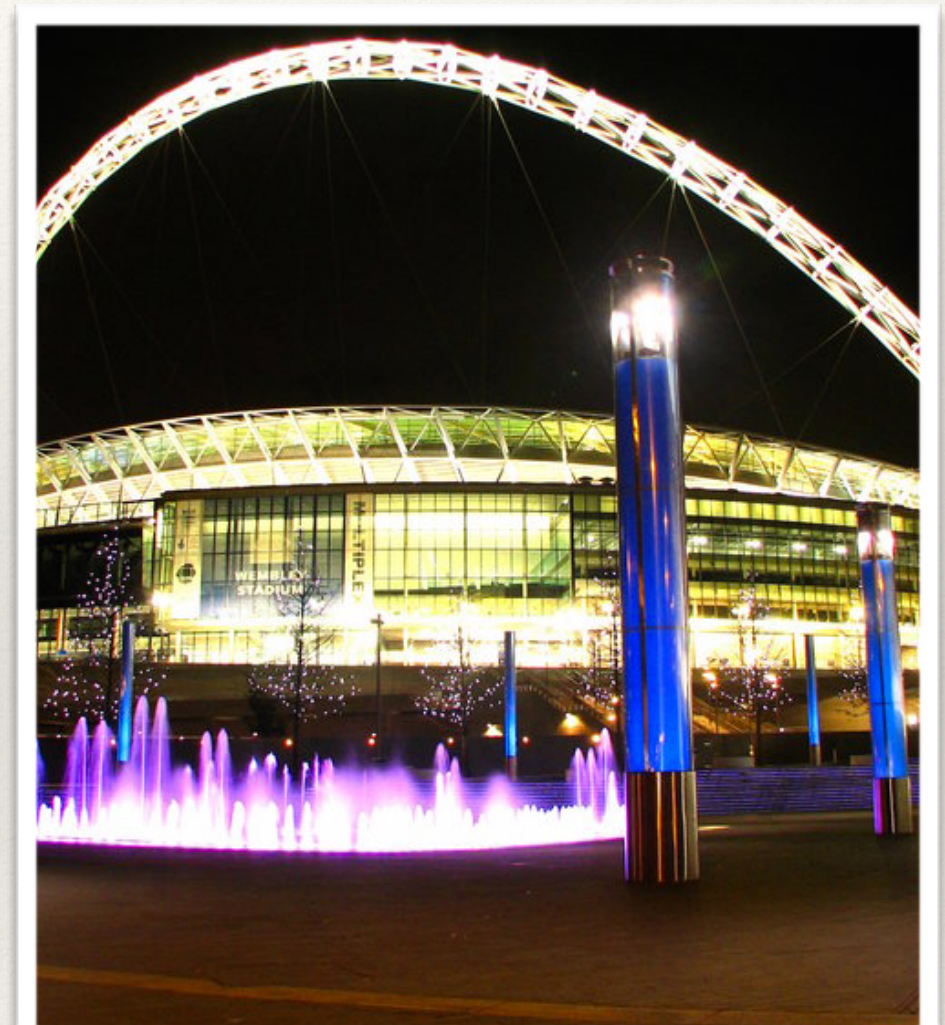
Joint work with Werner Nutt and Matthias Perktold





# Construction Projects are “Rarely” On-Time/On-Budget

- Wembley National Stadium
  - Commenced: 2002
  - Planned Completion: Early 2006
  - Open: March 2007
  - *Delay: 1 year*
  - Planned Cost: £757m
  - Final Cost (approx.): £1bn
  - *Increased cost: 32%*



<http://www.globalconstructionreview.com/>



# Construction Projects are “Rarely” On-Time/On-Budget

- Berlin Airport
  - Commenced: 2006
  - Planned Completion: ~~2011~~ ~~2012~~ ~~2013~~
  - Open: **Oct 2020 ???**
  - *Delay: 9 Years*
  - Planned cost: €2bn
  - Current Cost: €7bn
  - *Increased cost: 200%*



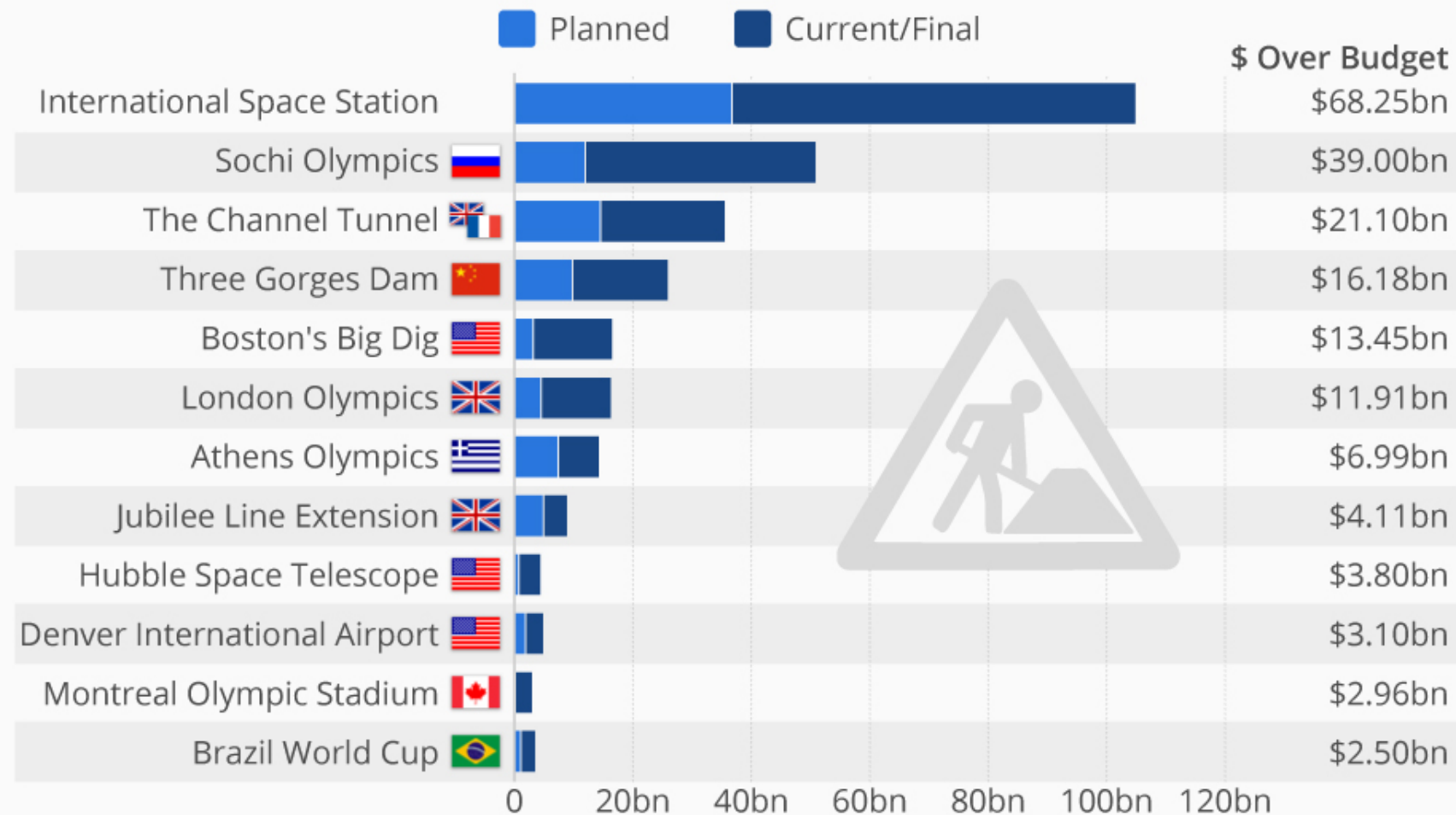
<https://www.economist.com/the-economist-explains/>



# ...and Many More

## Major Projects That Went Catastrophically Over-Budget

Selected over-budget construction projects worldwide (billion U.S. dollars)\*



CC BY ND  
@StatistaCharts

\* Converted to U.S. dollars and adjusted for inflation.  
Source: Podio.com

statista



# What are the challenges

- Coordination
  - Between SMEs
  - With the Supply Chain

- ≠ Manufacturing
- ≠ Admin Proc.



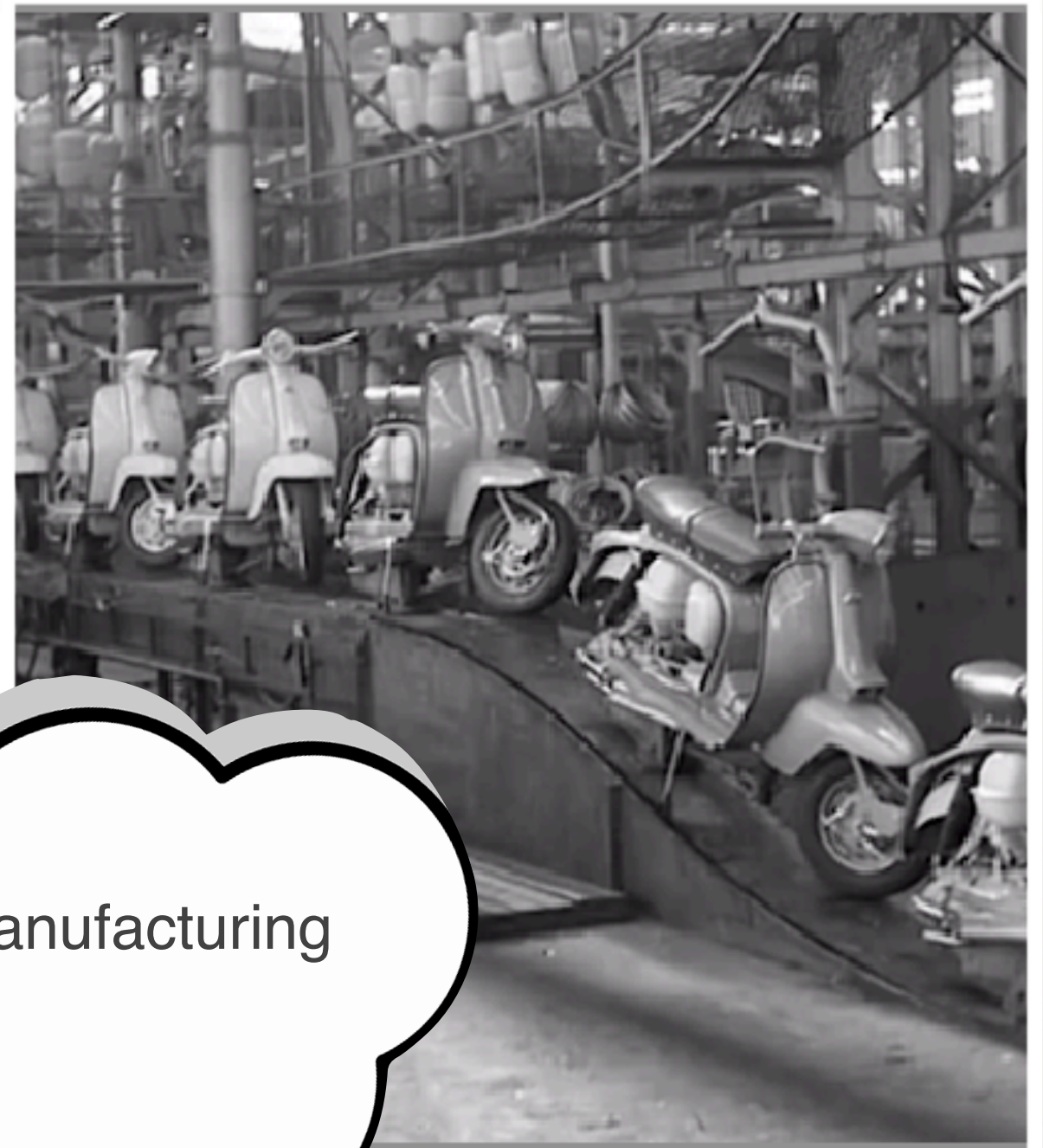


# What are the challenges

- **Coordination**
  - Between SMEs
  - With the Supply Chain
- **Low Standardisation**
  - Each project is one of a kind

- SMEs Consortium
- Technically

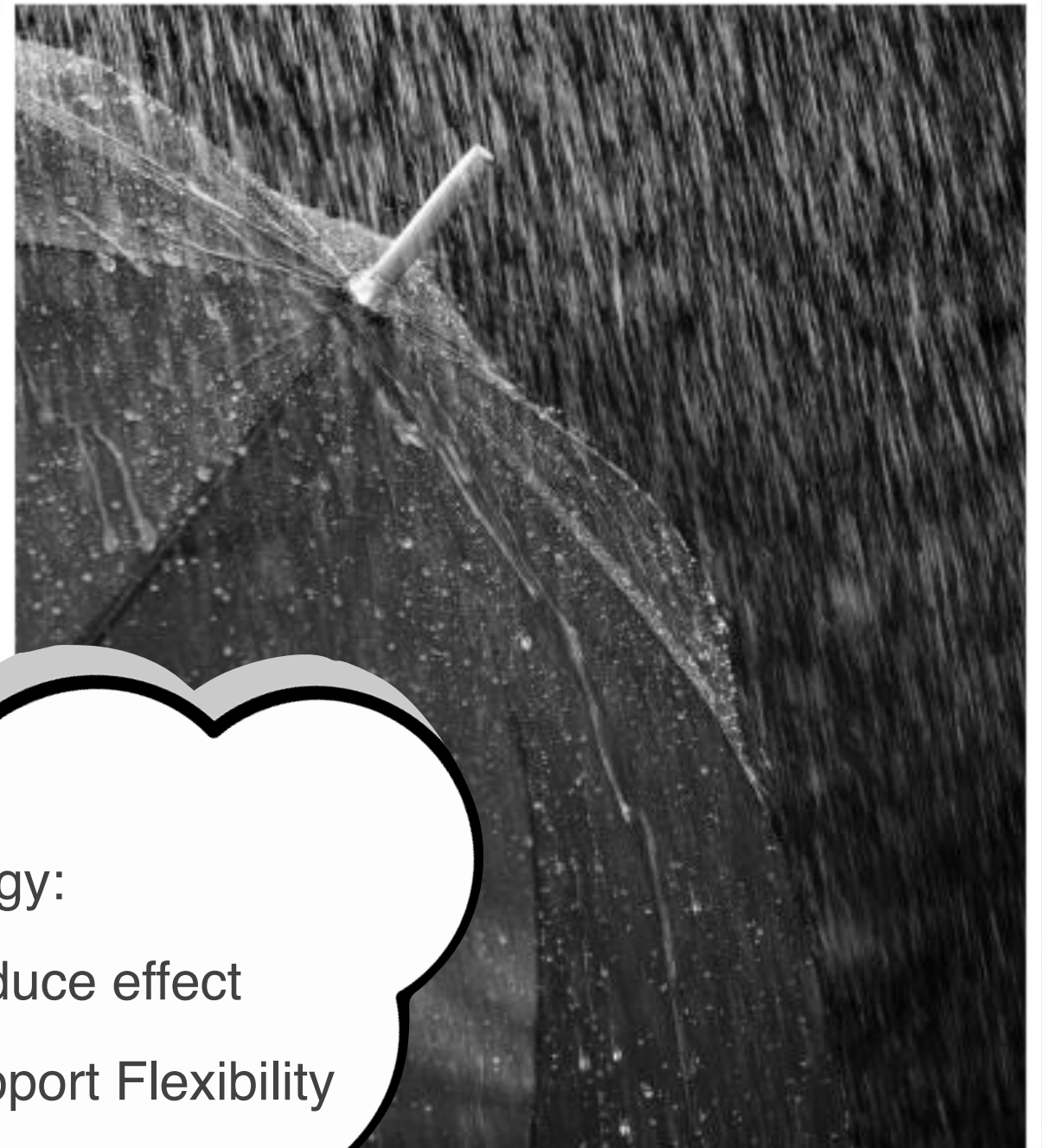
- $\neq$  Manufacturing





# What are the challenges

- **Coordination**
  - Between SMEs
  - With the Supply Chain
- **Low Standardisation**
  - Each project is one of a kind
- **Imponderabilities**
  - E.g., weather conditions, changing requirements
  - Unavoidable

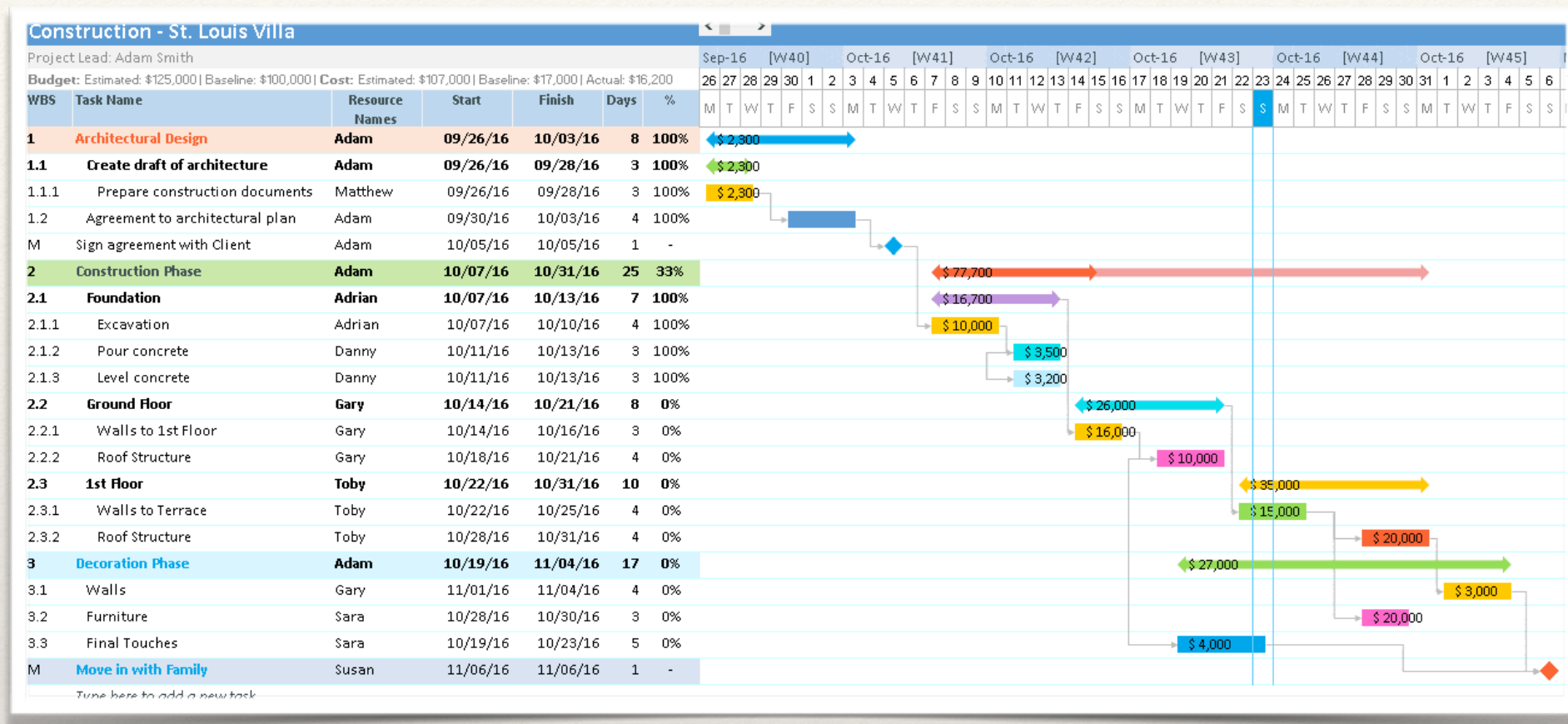


Strategy:

- Reduce effect
- Support Flexibility



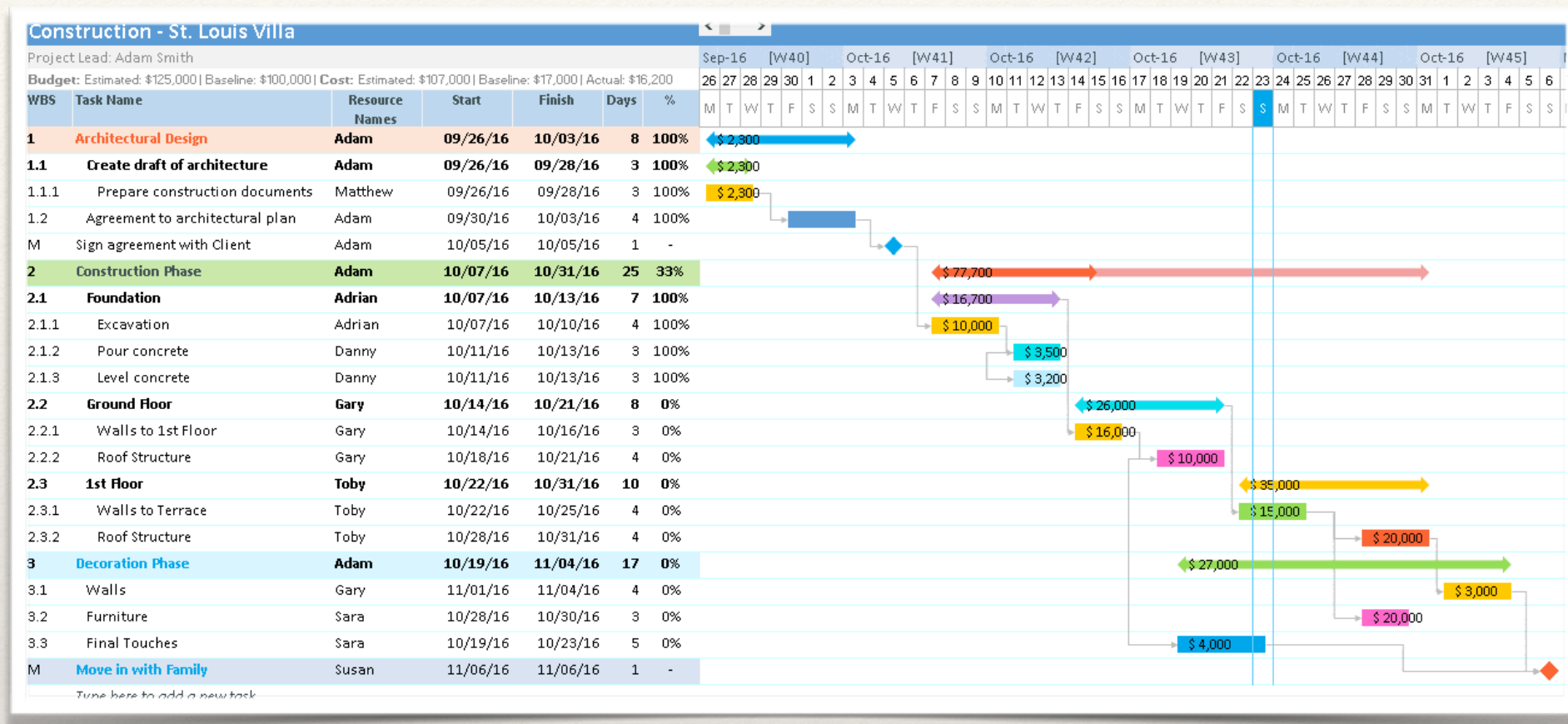
# Current Approach



- Execution Process Management: **Gantt Charts**
- Defined by the Project Manager
- Tools: MS Project / MS Excel



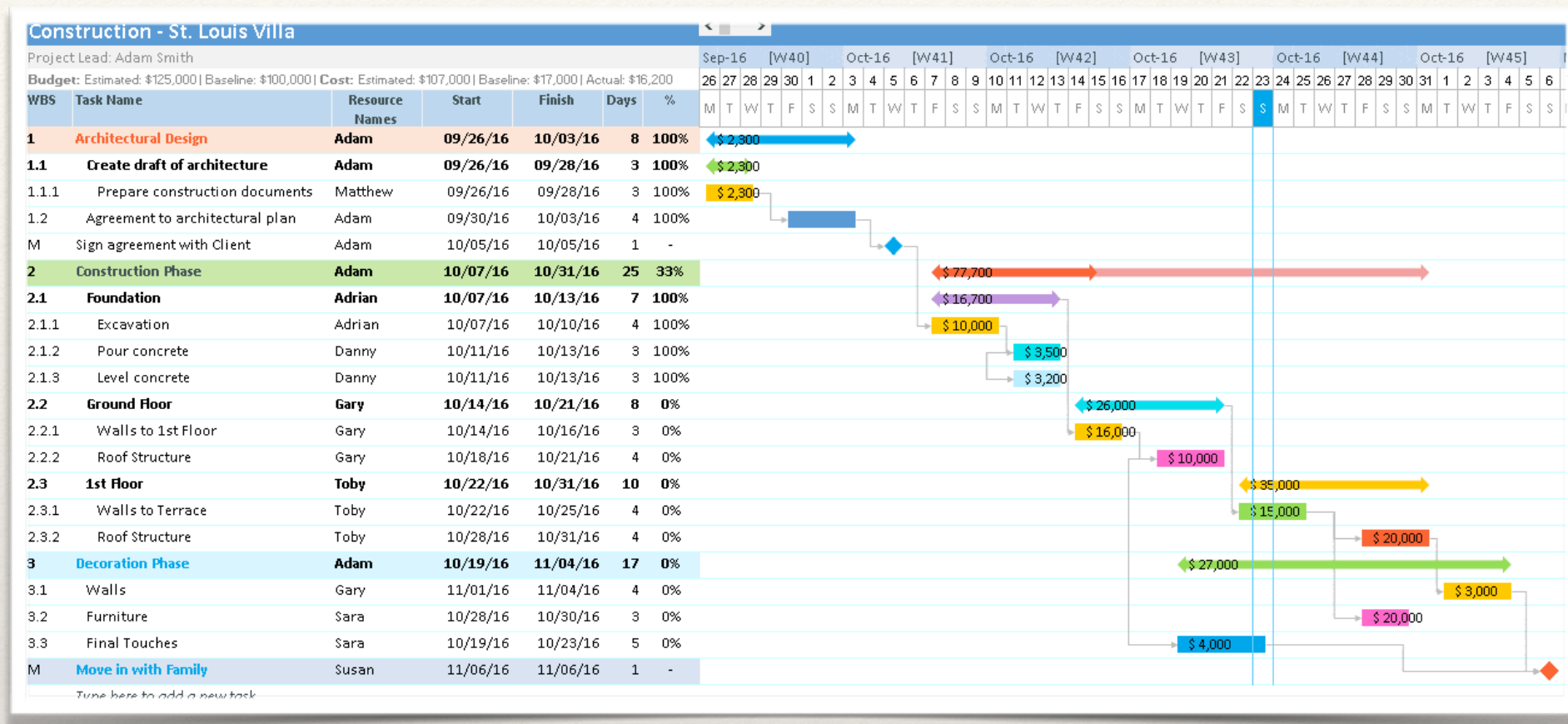
# Gantt Charts: Pros



- For the entire **duration** of the project
- Show alternation of companies on-site
- Show **milestones**
- Support communication with the **customer**



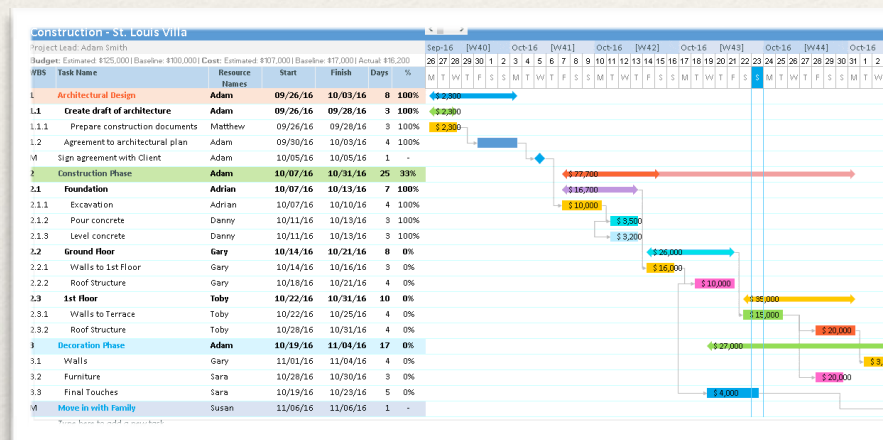
# Gantt Charts: Cons



- General purpose: no proper **abstractions** (e.g., Locations, Precedences)
- Difficult to **update**
- Not **detailed** for a daily / weekly schedule
- They already represent a (long-term) commitment



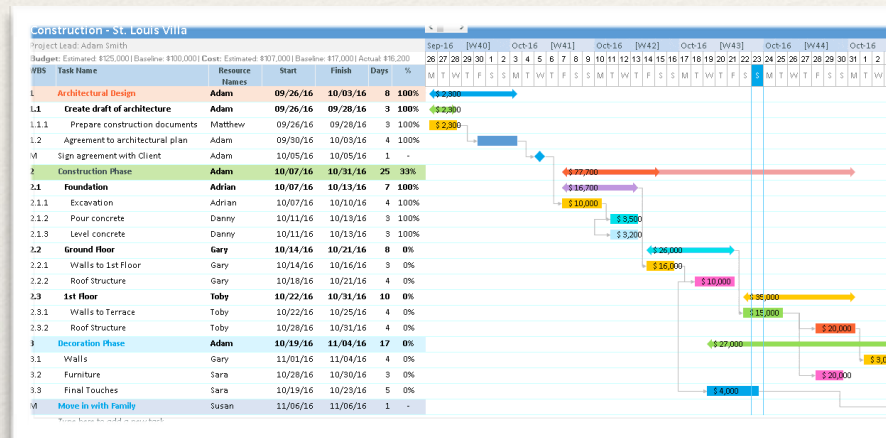
# Management with Gantt Charts (1)



- There is no clear definition of the **process requirements**:
  - **Single point of failure**: PM / Foreman
  - Alternative plans? Optimisation? Automation?



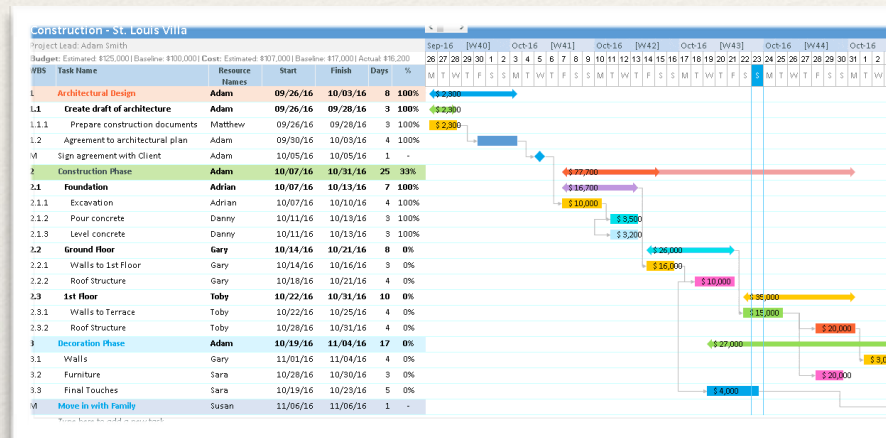
# Management with Gantt Charts (2)



- Daily / Weekly schedule defined “on the fly”
  - No short- and **medium-term planning**
  - Synch with supply chain and between companies



# Management with Gantt Charts (3)



- Do not support **report** on the **actual progress**
  - Progress estimated based on indirect measures
  - Are used for tendering purposes only (**no updates**)



---

# COCKPiT

---

- COCKPiT  
Collaborative Construction Process Management
- Applied research project
- European Regional Development Fund  
(ERDF -EFRE - FESR1008) of the  
Autonomous Province of Bolzano-South Tyrol



# Scientific Team

**unibz** Fakultät für Informatik  
Facoltà di Scienze e Tecnologie informatiche  
Faculty of Computer Science

**unibz** Fakultät für Naturwissenschaften und Technik  
Facoltà di Scienze e Tecnologie  
Faculty of Science and Technology

**Fraunhofer**  
ITALIA



Werner Nutt



Elisa Marengo



Patrick Dallasega



Carmen Marcher



Mehtab Alam



Peng Cheng



Hebatallah  
Mohamed



Dominik Matt



Andrea Revolti



Christoph Schimanski



Camilla Follini



# Companies



## Interior

- Environmental engineering
- Water technology
- Energy engineering
- Building services



## Envelope

- Engineering
- Production
- Installation of facades



## Skeleton

- Competences as bricklayers, carpenters and metalworkers
- Acting also as General Contractors



# COCKPiT: What is it?

- Objective:  
Improve Execution Process Management in Construction by
  - Developing **Methodologies**
  - Increasing **Digitalisation**

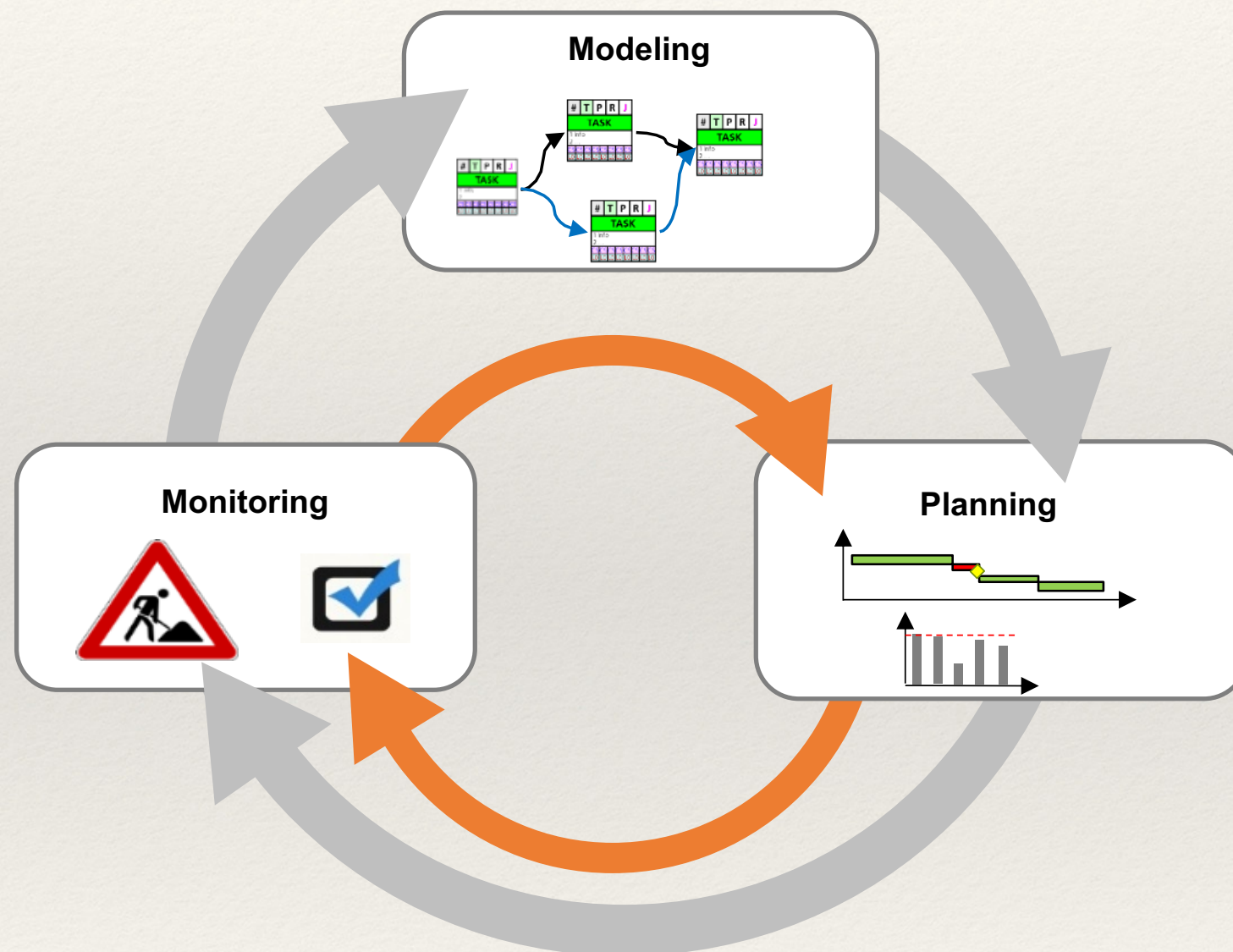


# COCKPiT: What is it?

- Objective:  
Improve Execution Process Management in Construction by
  - Developing **Methodologies**
  - Increasing **Digitalisation**
- This would allow:
  - reduce delays
  - reduce cost overruns
  - better synchronisation with the supply chain
  - better usage of the resources
  - ...



# How To Do That





# Topic of Today: Modelling

- Objective: Explicit representation of the **process requirements**
  - > No more single point of failure
  - > Identification of alternative plans
  - > Automation / Optimisation
  - > Flexibility in handling imponderabilities



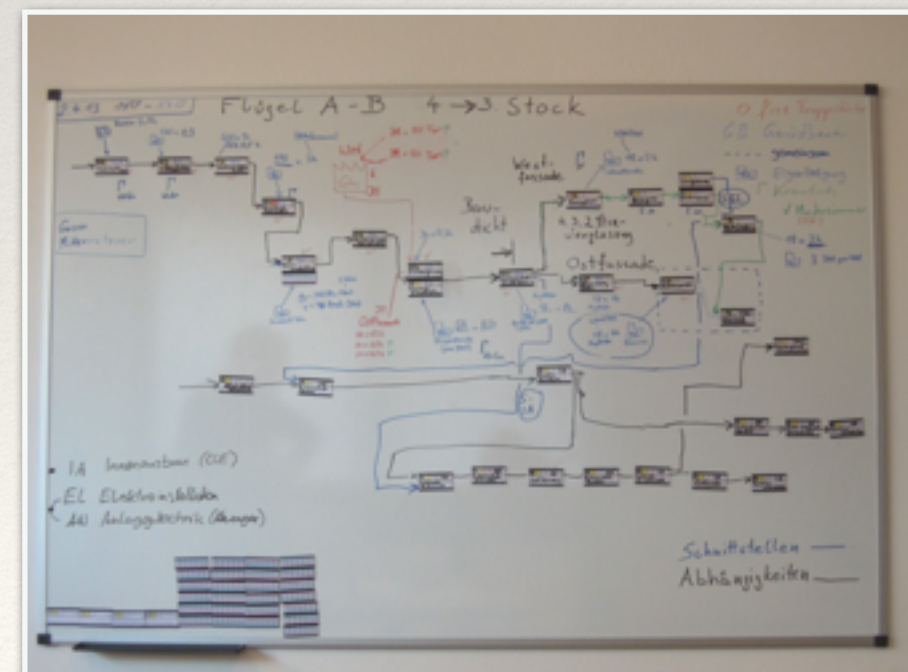
# Topic of Today: Modelling

- Objective: Explicit representation of the **process requirements**
  - > No more single point of failure
  - > Identification of alternative plans
  - > Automation / Optimisation
  - > Flexibility in handling imponderabilities
- **Requirements:**
  - Proper abstractions
  - Formal approach



# First Approach

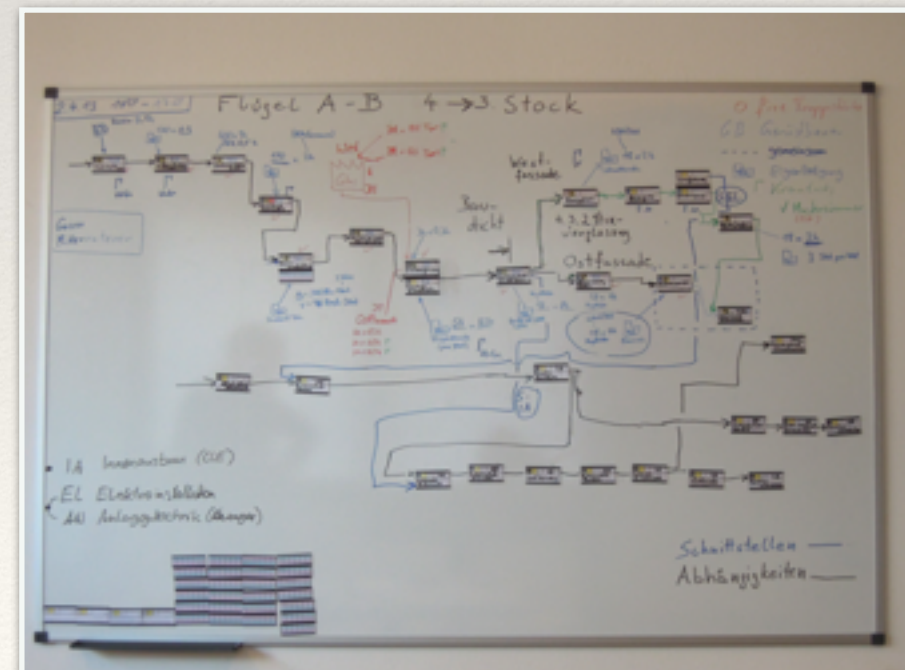
- Expansion of the Bolzano Hospital
- Modeling
  - Decoupled from scheduling
  - Define **What** and **Where** (not yet when)
  - Collaborative modelling





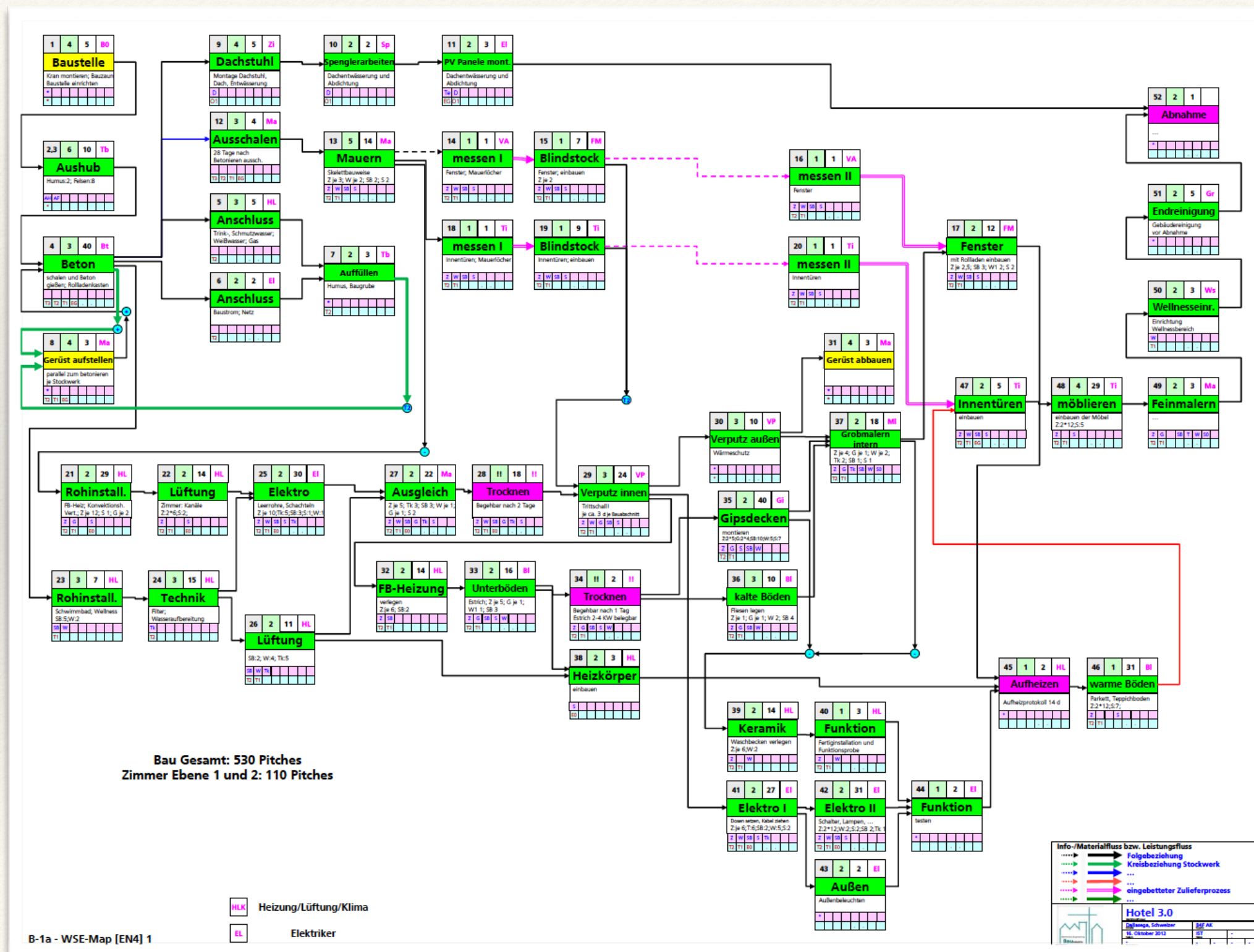
# First Approach

- Expansion of the Bolzano Hospital
- Modeling
  - Decoupled from scheduling
  - Define **What** and **Where** (not yet when)
  - Collaborative modelling
- Scheduling
  - Define **How** and **When**





# Bolzano Hospital





# Elements in the model

- Tasks
  - What: Activity
  - Who: Craft
  - Where: Locations
  - How long: Productivity
  - Notes

Productivity

#184	2	5	FL
LF - Lay Floor			
Z	G	W	
T1	T2	*	

Craft

Activity

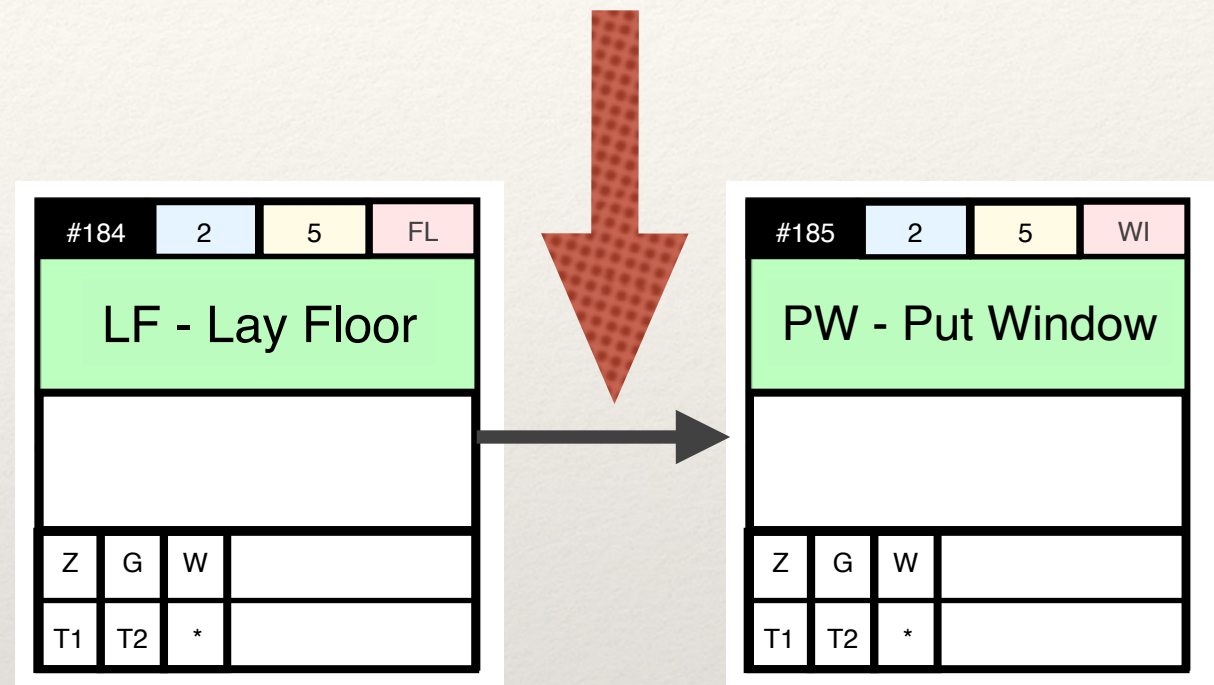
Notes

Locations



# Elements in the model

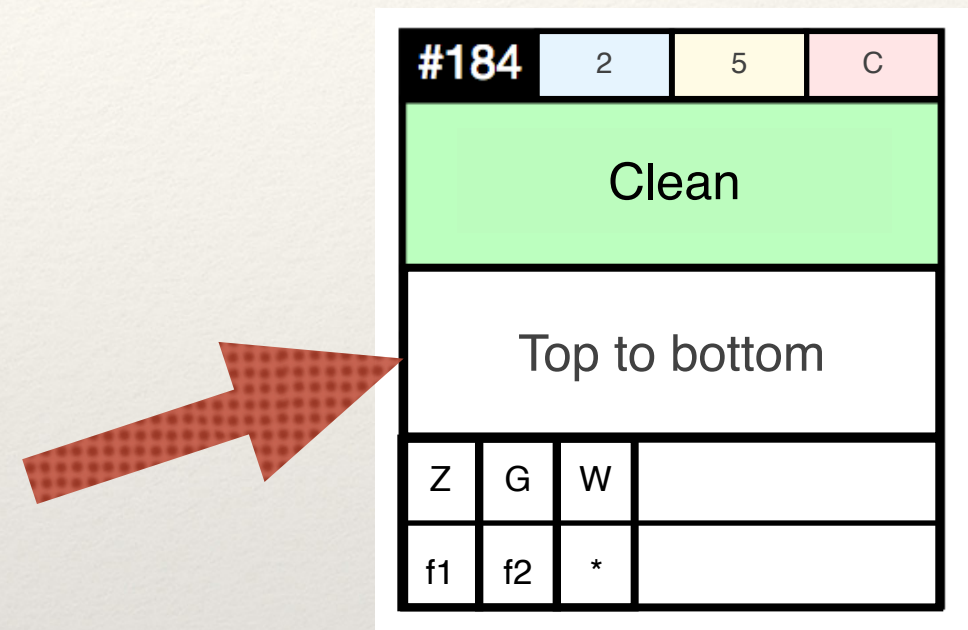
- Tasks
  - What: Activity
  - Who: Craft
  - Where: Locations
  - How long: Productivity
  - Notes
- Synchronisation
  - Declarative precedences





# Hidden Knowledge and Ambiguities

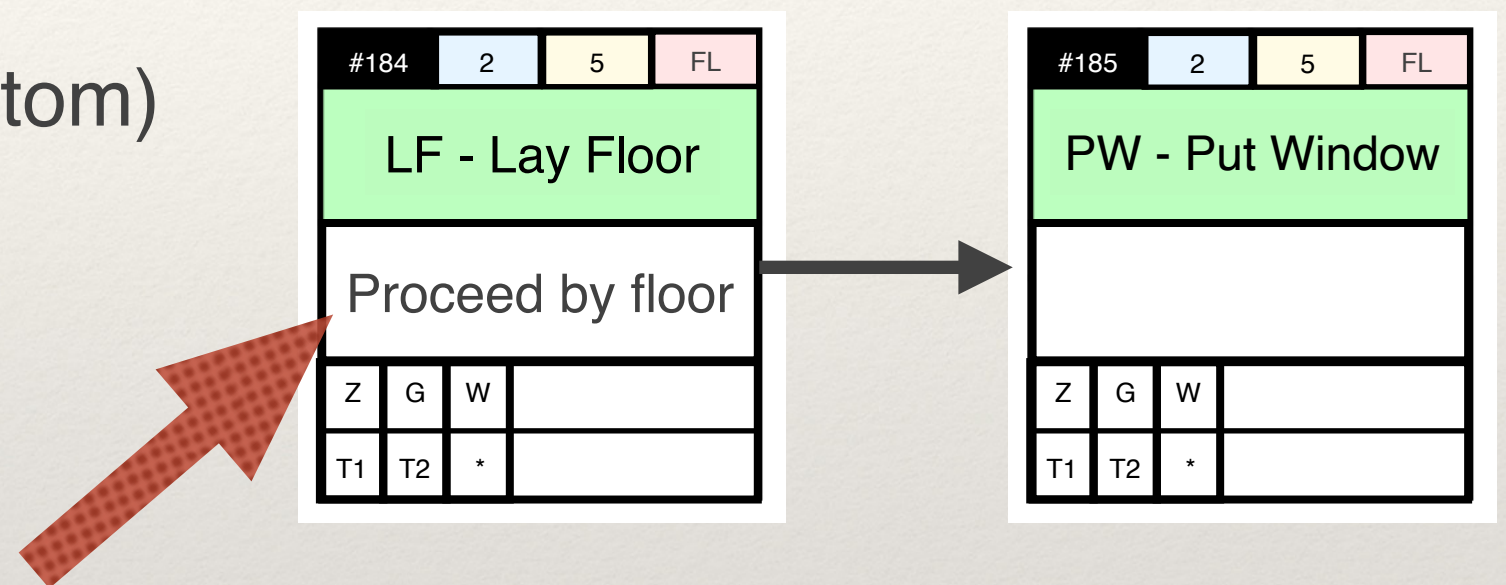
- **Orderings**  
among the locations  
(bottom to top, top to bottom)





# Hidden Knowledge and Ambiguities

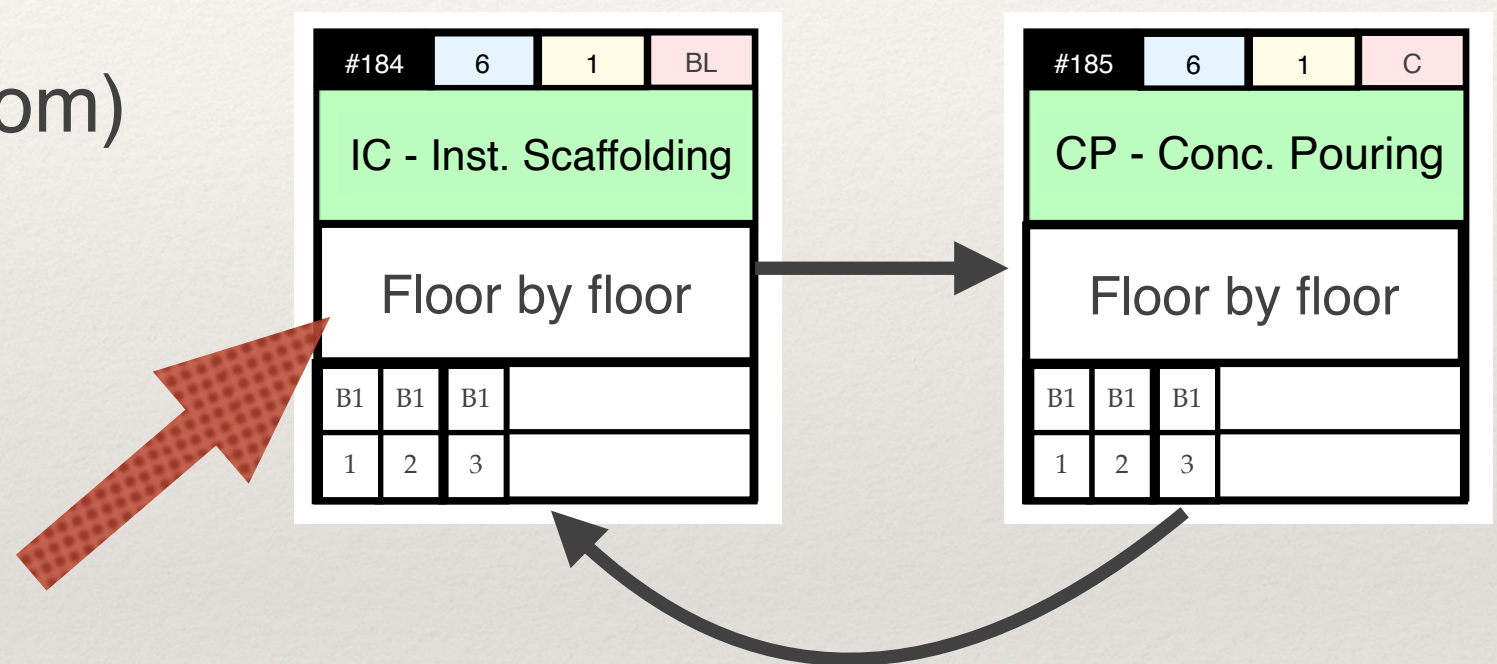
- **Orderings**  
among the locations  
(bottom to top, top to bottom)
- Precedence **Scope**  
(floor, activity, building)





# Hidden Knowledge and Ambiguities

- **Orderings**  
among the locations  
(bottom to top, top to bottom)
- Precedence **Scope**  
(floor, activity, building)
- How to perform **Loops**





# Topic of Today: Modelling

- Objectives
  - > No more process requirements



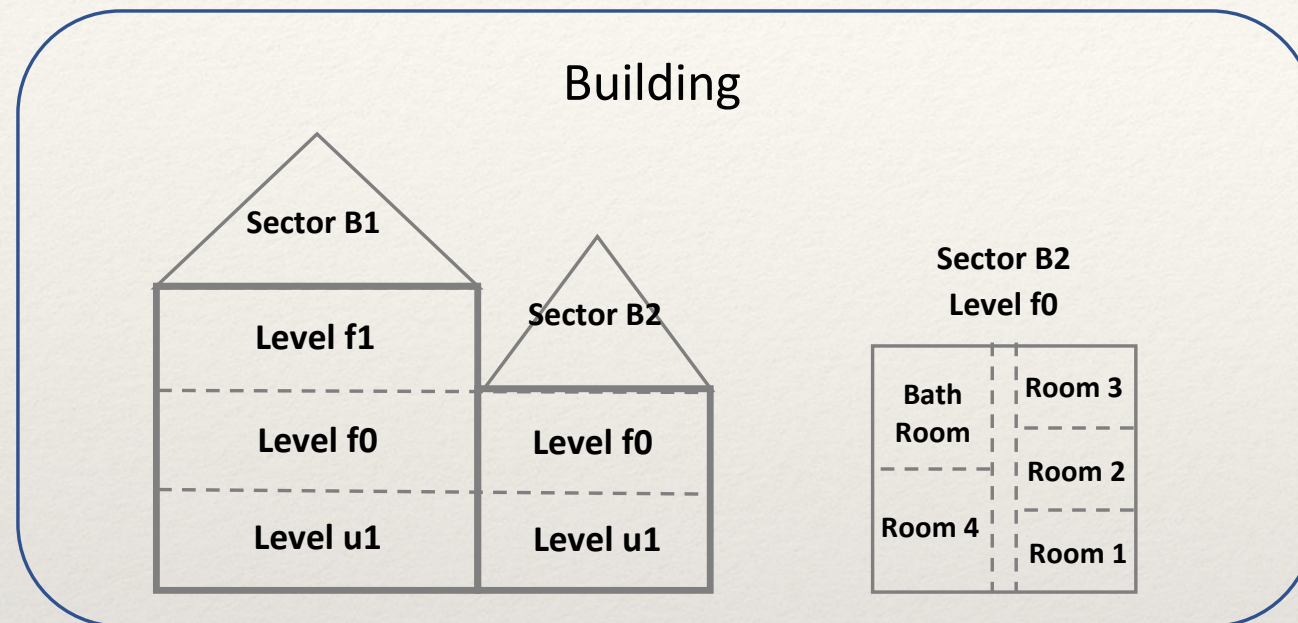
## CoPMod: Construction Process Modeling

- We extended the language (inspired by Declare)
- Provide a logic-based (LTLf) semantics

- Formal

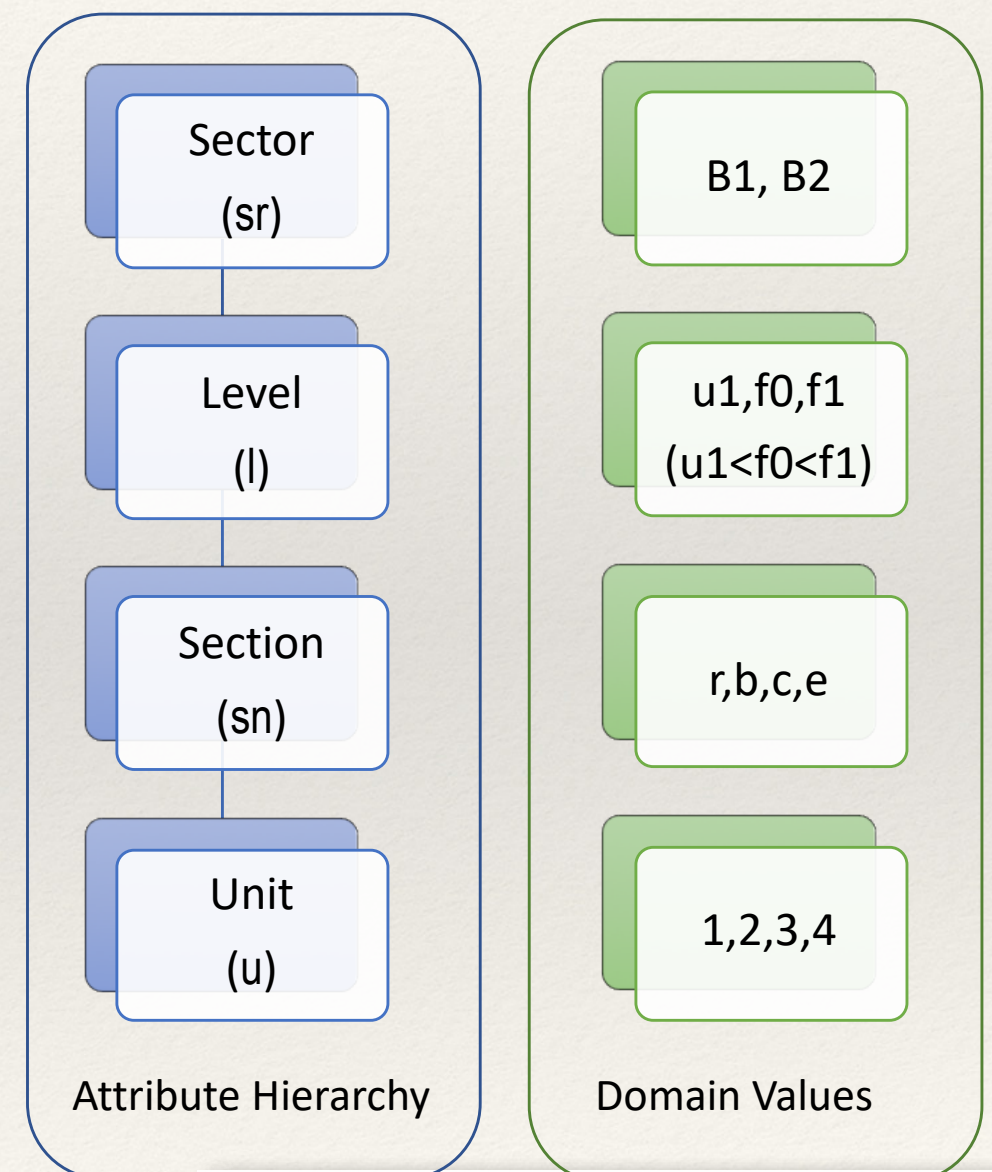
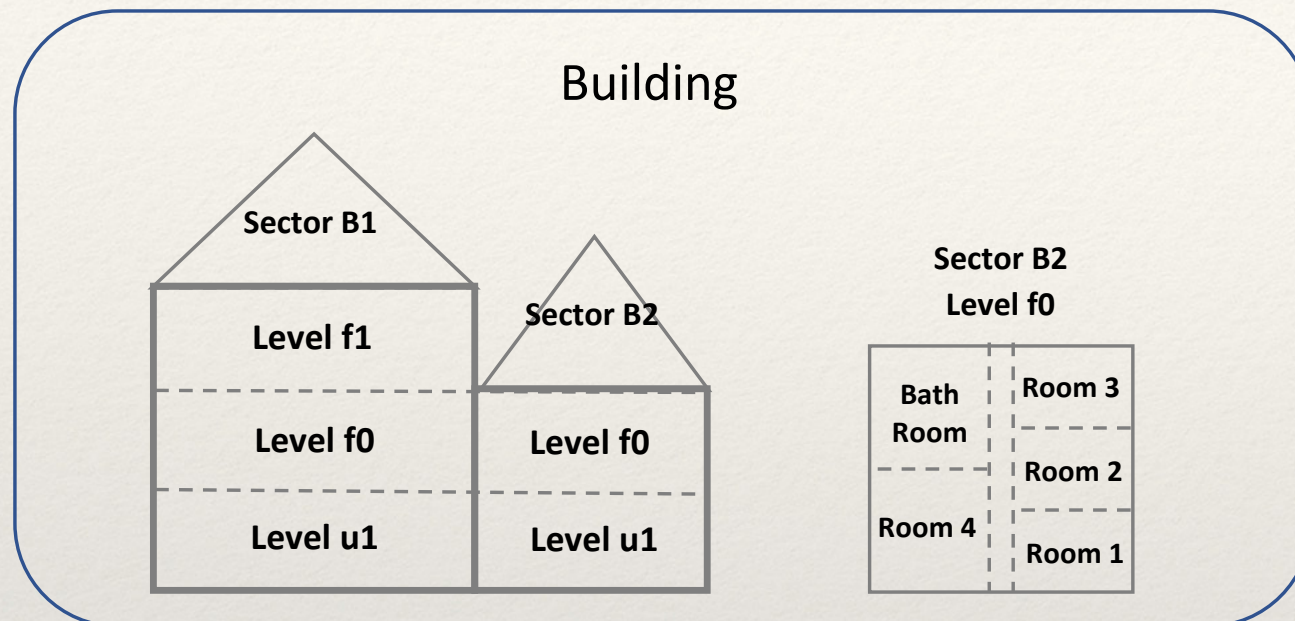


# Customisable Building Representation



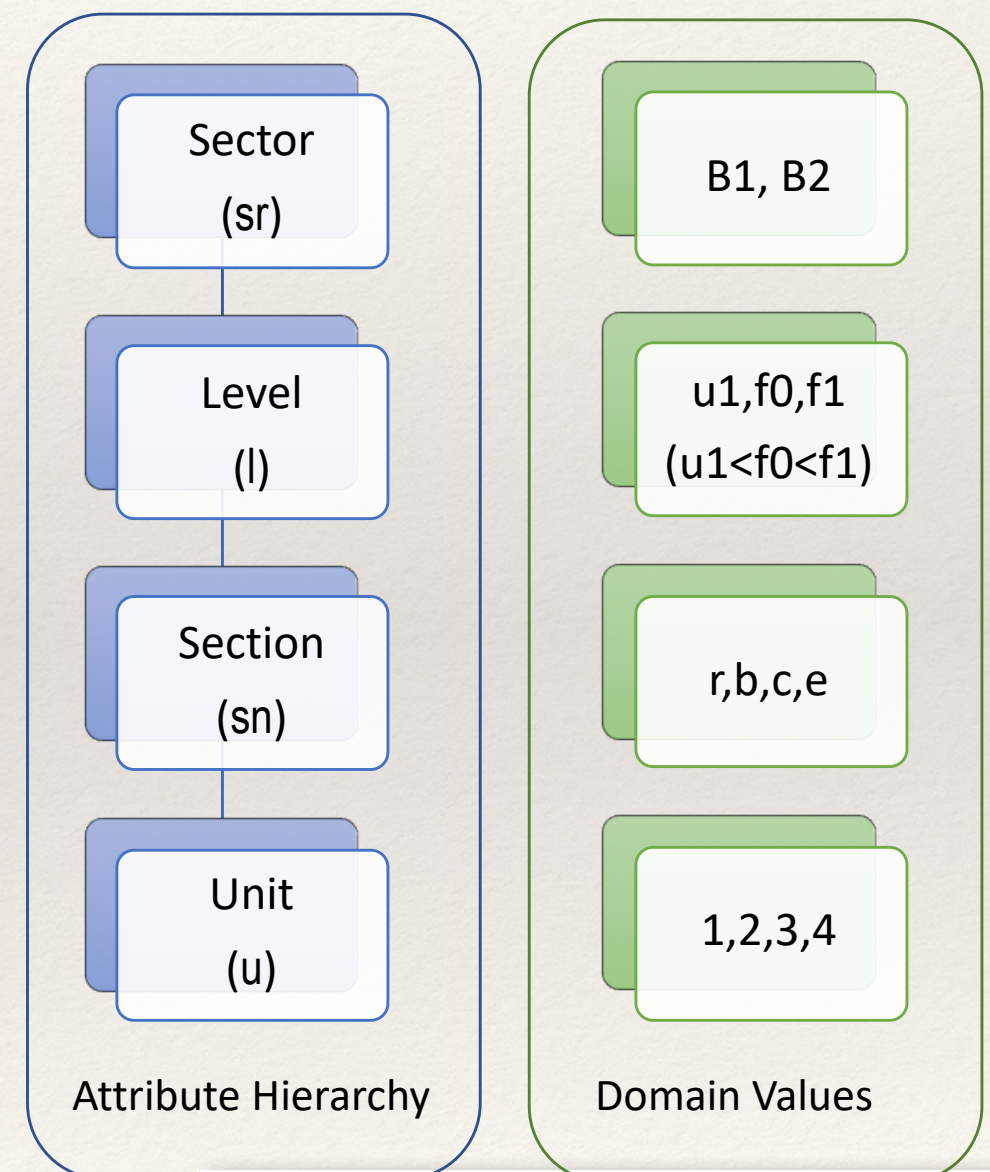
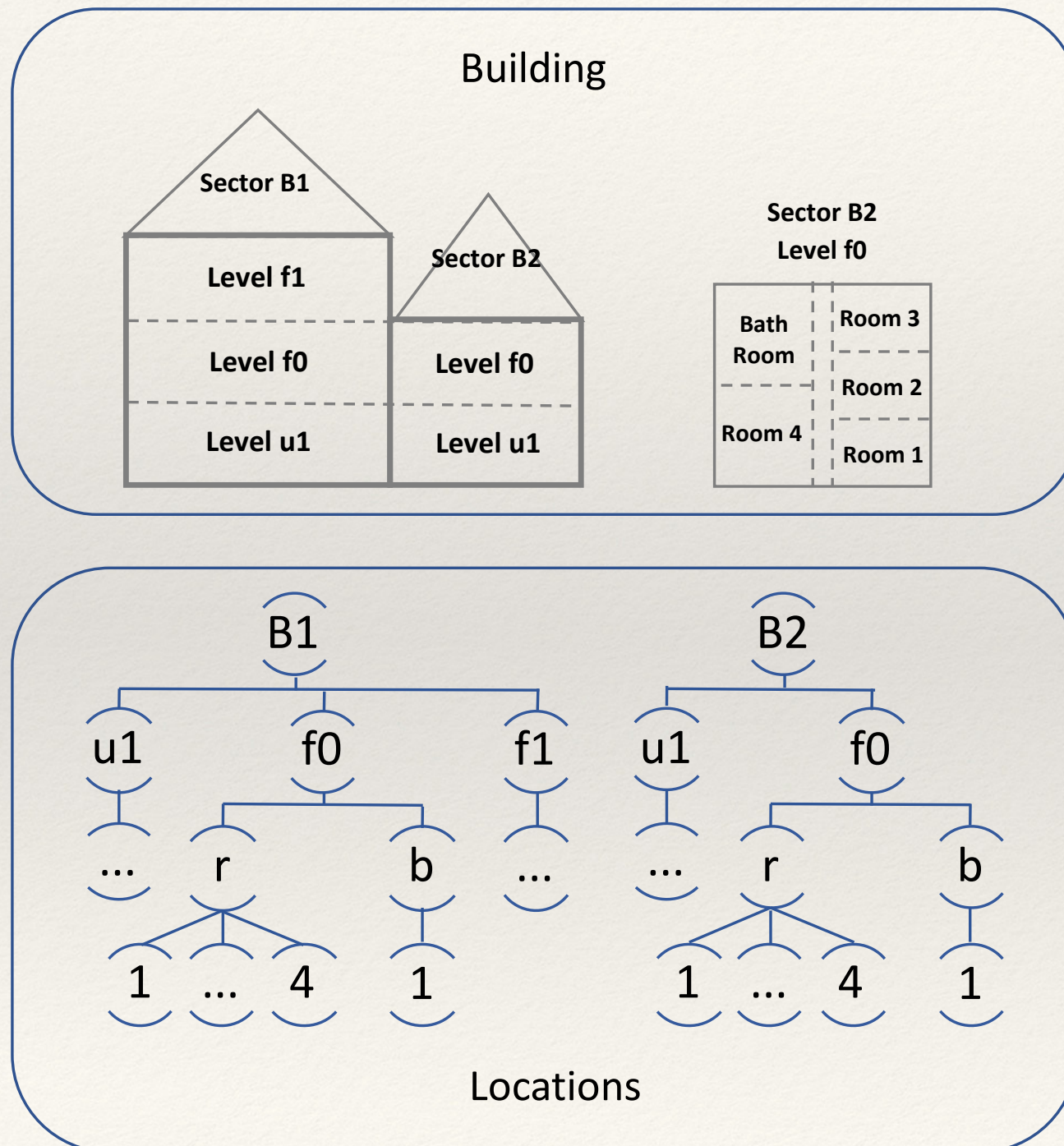


# Customisable Building Representation





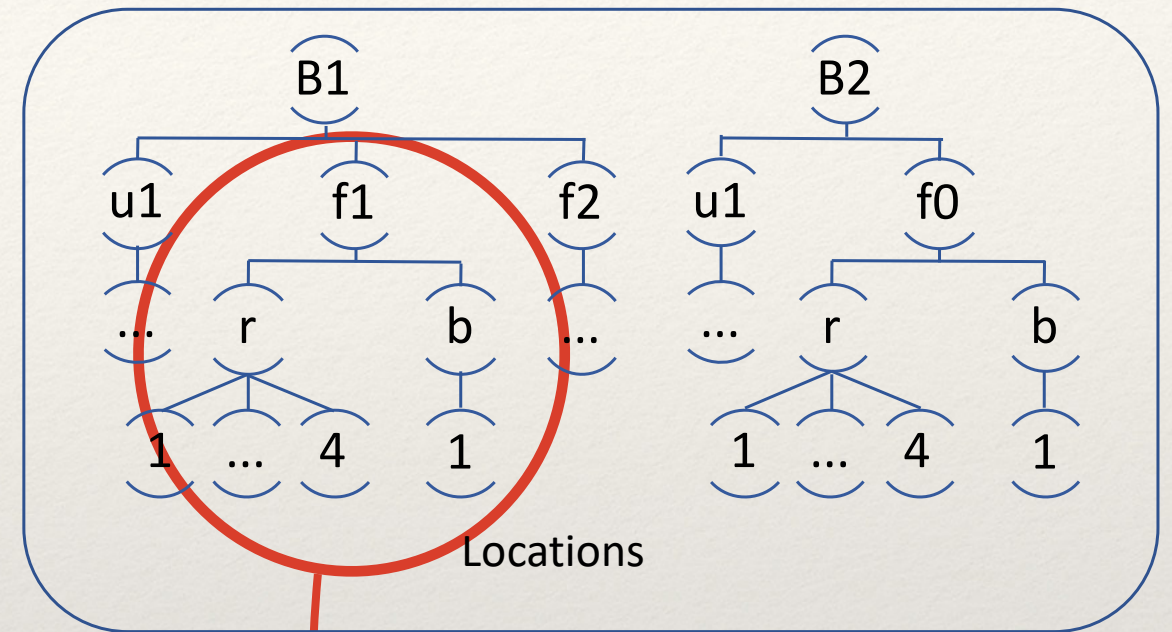
# Customisable Building Representation





# Representation of Locations

- A building is abstractly represented as a tree
- Locations in the tasks are **subtrees**



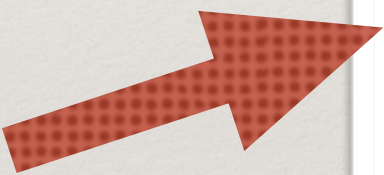
$\langle B1, f1, r, * \rangle$   
 $\langle B1, f1, b, * \rangle$

#113	1x2	60d	FL
LF - Lay Floor			
< : NONE		ex:	sr,l
B1	B1	B1	B2
f0	f1	f1	f0
e	r	b	r
1	*	*	*



# Ordering Constraints

- Attribute domain values can be ordered
- **Ascending** and **descending** ordering constraints



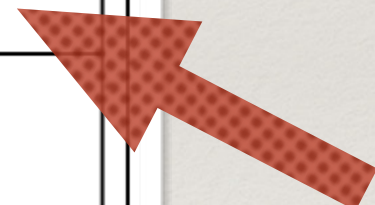
#148	1x1	20d	Sc
SI - Scaffolding Installation			
<:  ↑		ex: UNIT	
B1	B1	B2	
f0	f1	f0	



# Exclusivity Constraints

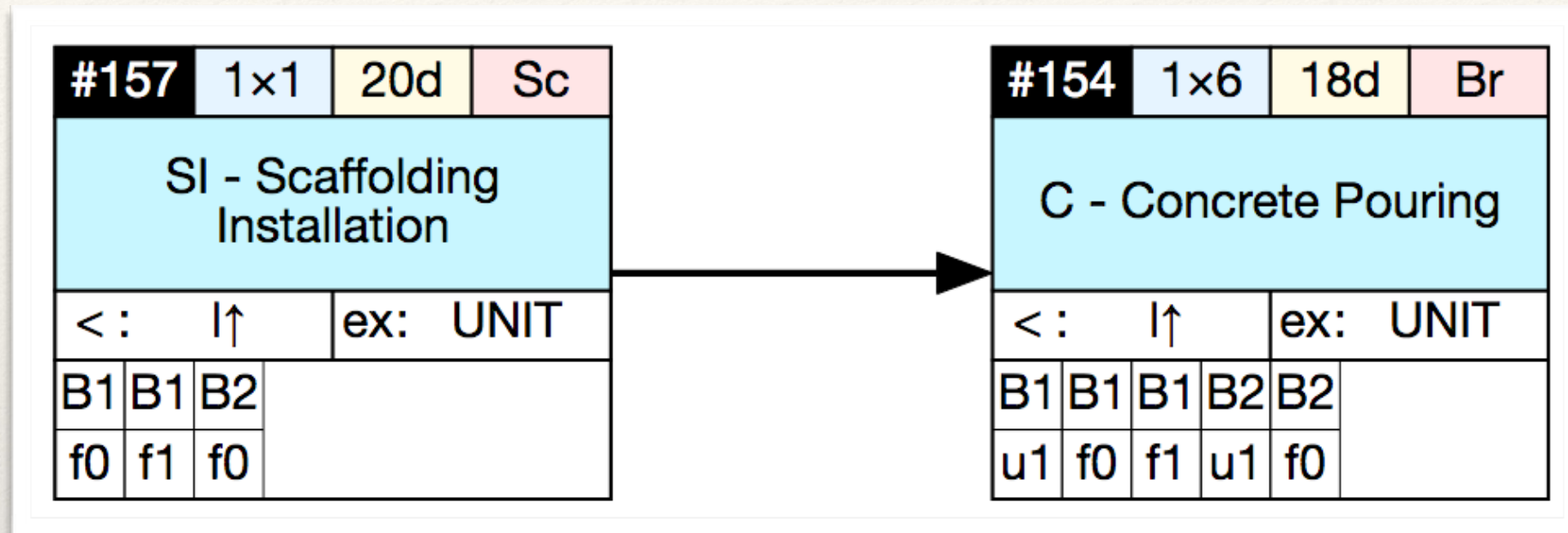
- Once the task is started, **no other task** can be performed there
- By default: exclusivity at the **unit level**

#143	1x6	10d	Di
Ex - Excavation			
< : NONE		ex:	sr
B1	B2		
u1	u1		





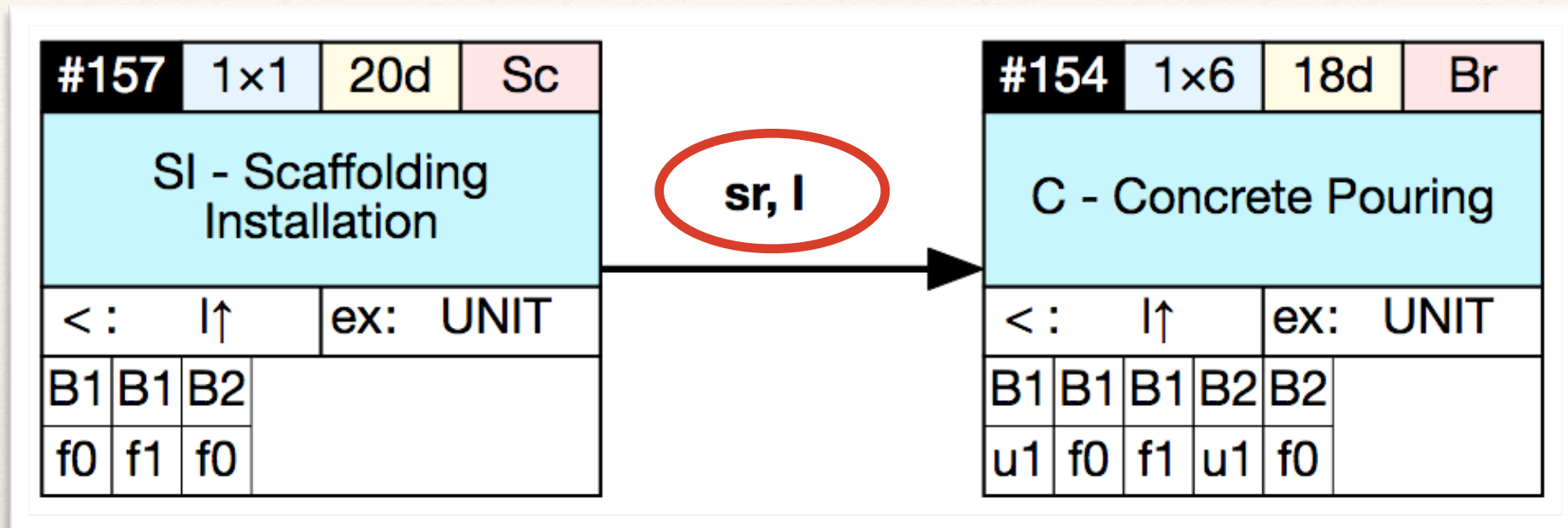
# Precedences



- Precedences between activities



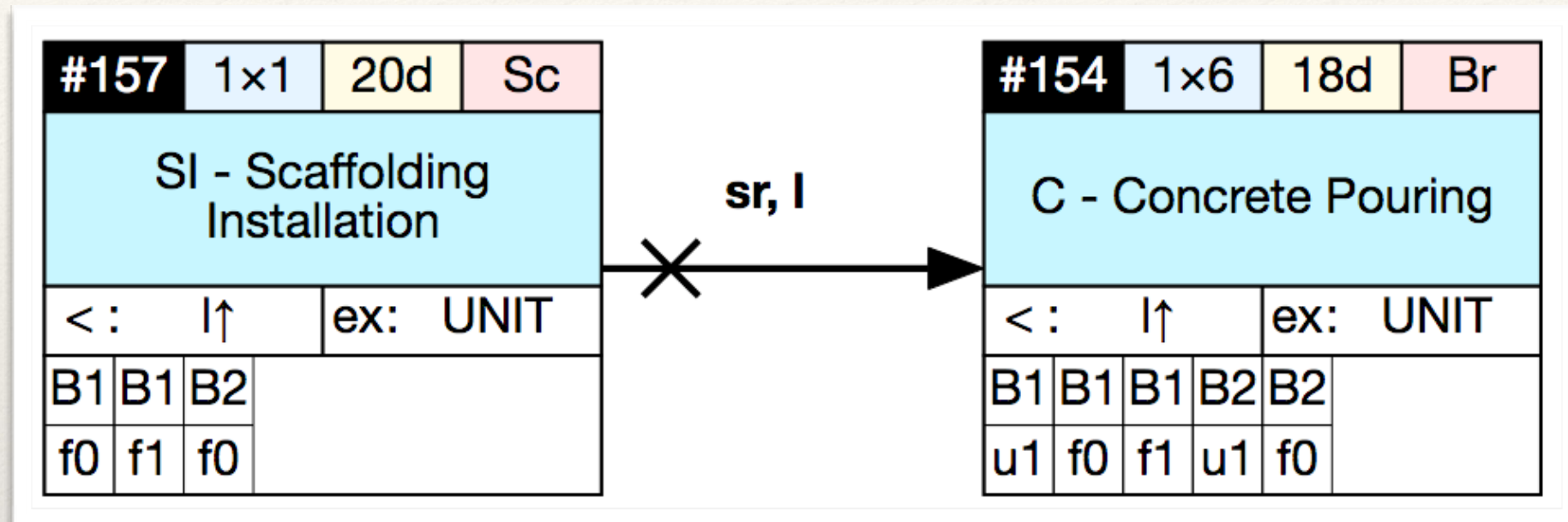
# Precedences: Scope



- The **Scope** specialises the precedence (e.g., precedence by <sector, level>)
- By default: **Activity level**



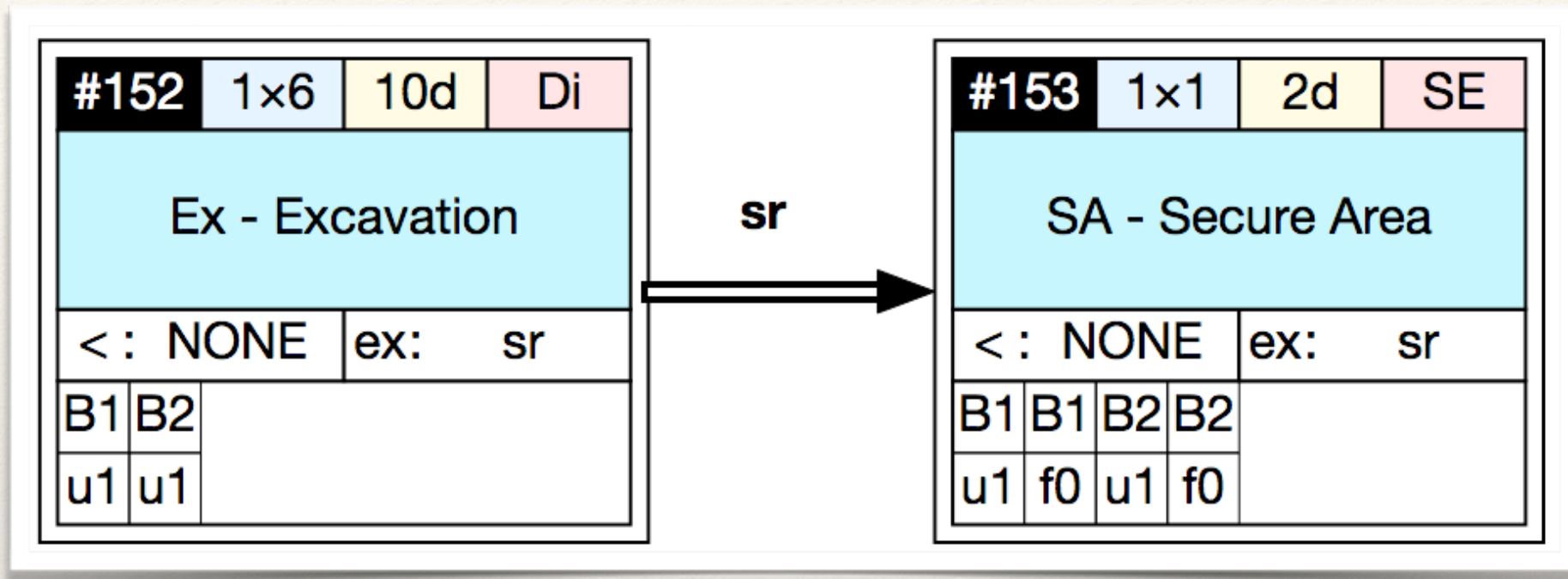
# Precedences: Alternate Precedence



- **Alternation** between antecedent and consequent:
  - antecedent **before** consequent
  - **and** the antecedent has to **wait** for the consequent



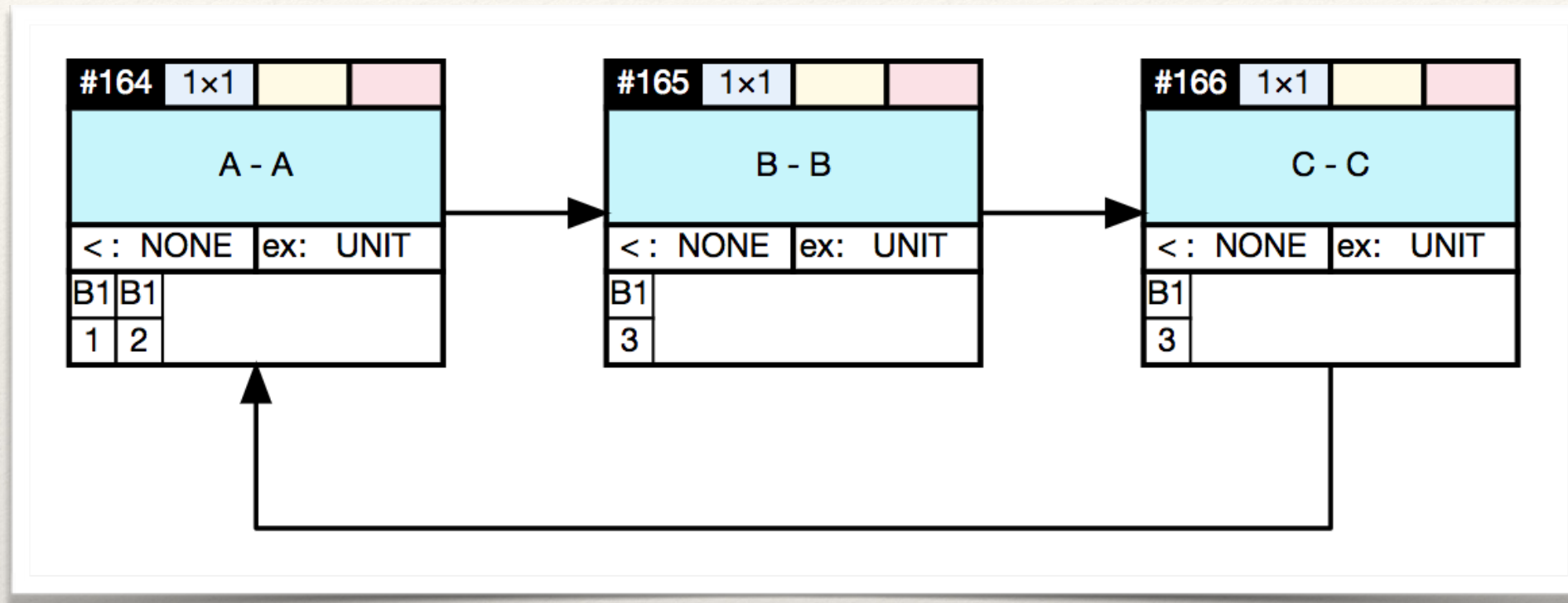
# Precedences: Chain Precedence



- **Chain** between two activities:
  - **no** other activities can be performed **in-between**

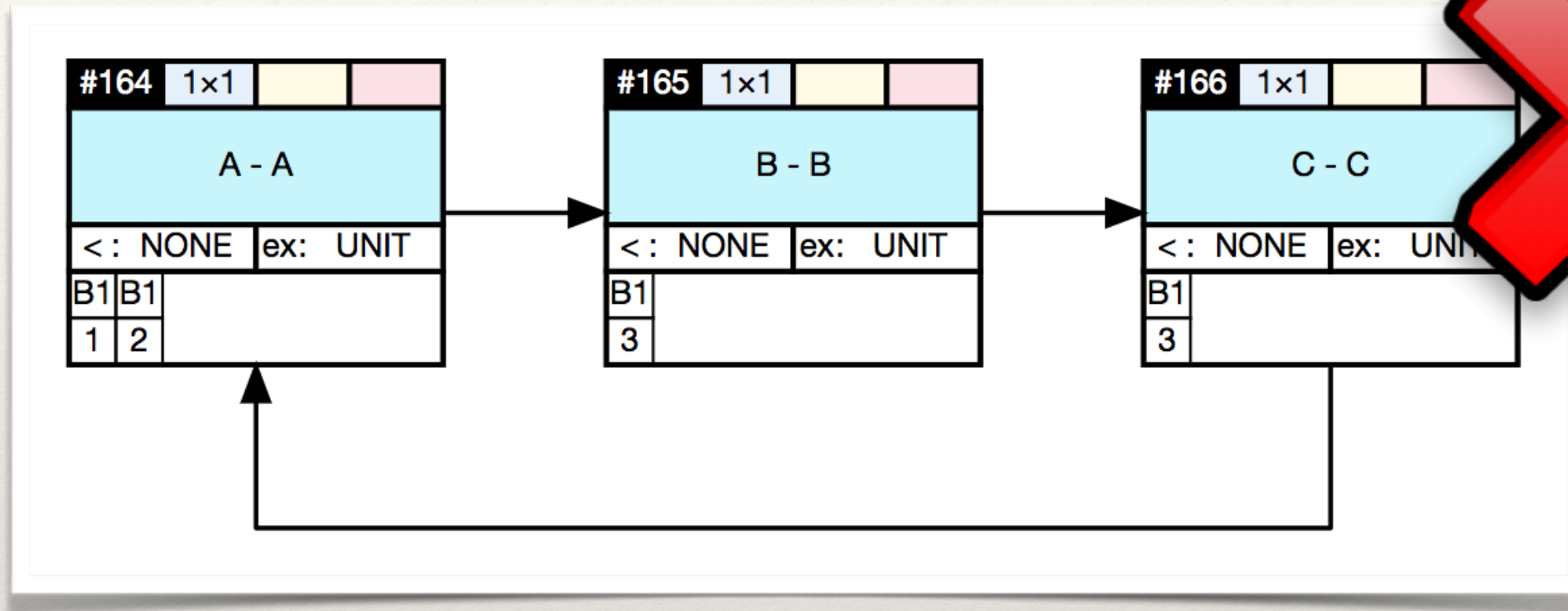


# Does my model make sense?





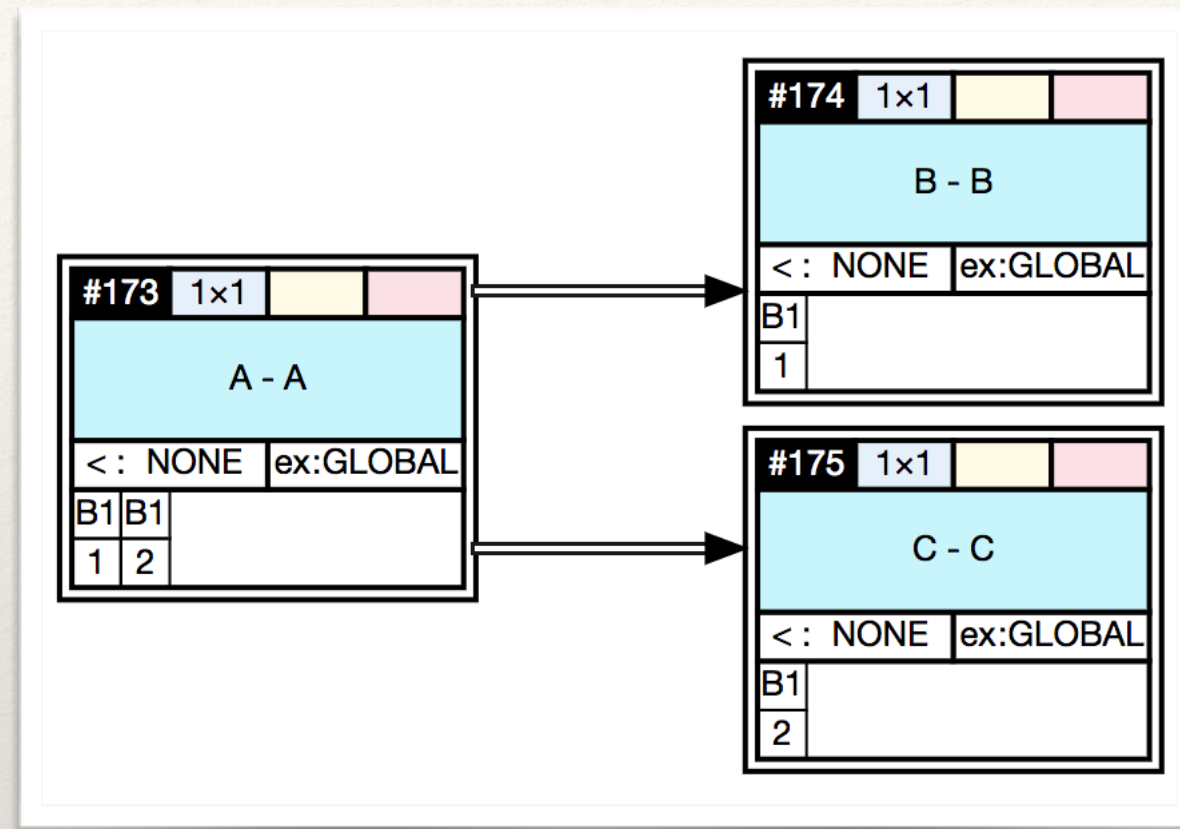
# Does my model make sense?



- Is there an execution satisfying all the constraints?  
**Satisfiability Check**



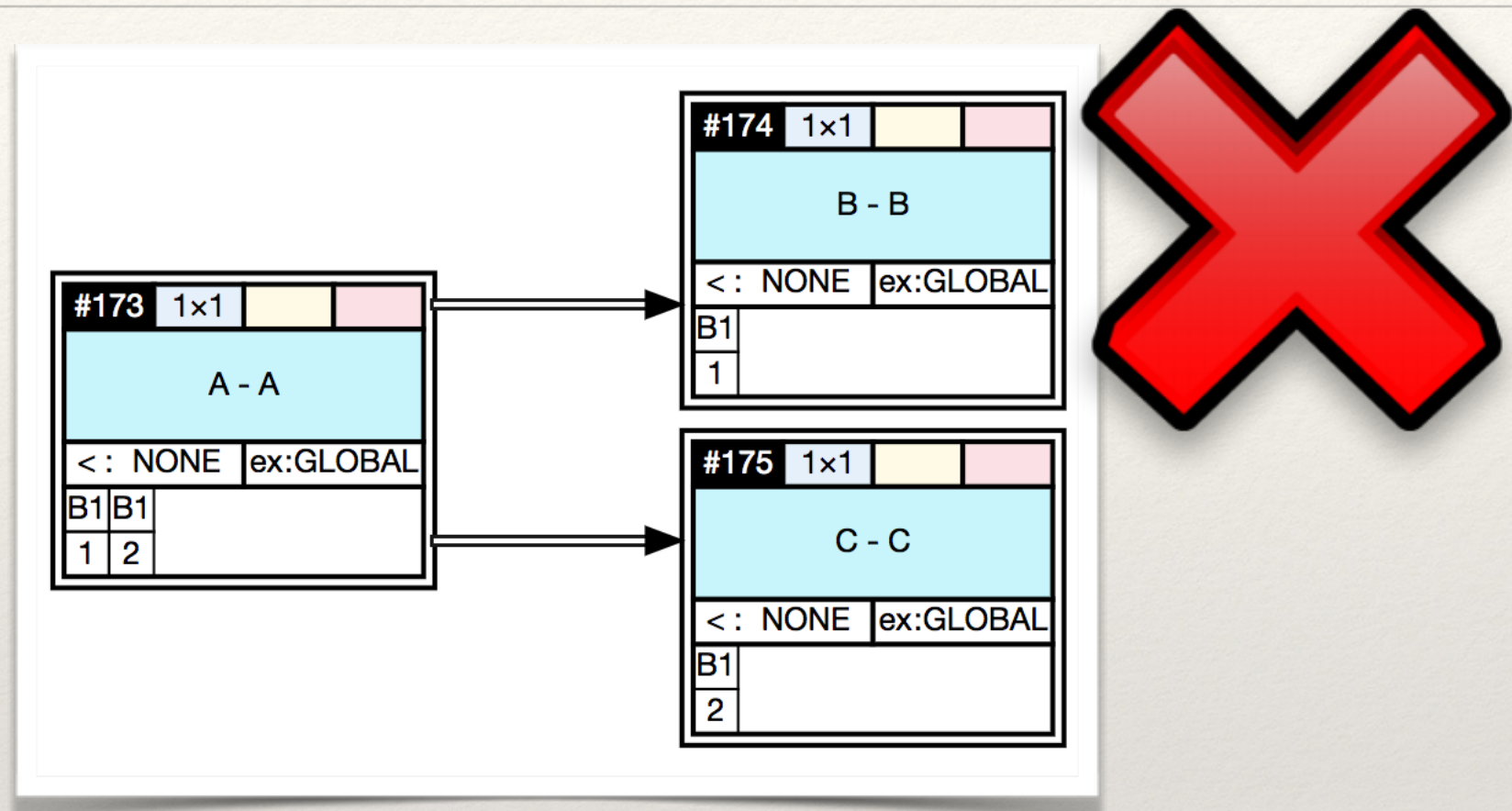
# Satisfiability Check



- Is checking for loops enough to determine **Satisfiability**?



# Satisfiability Check



- Is checking for loops enough to determine **Satisfiability**?
- **No,**
- Consider also the **dependencies**, **scopes** and **locations**



# How to Check Satisfiability?

- Our model has a logic based semantics (LTLf)
- We can apply model checking techniques
- We performed some experiments using NuSMV  
(state-of-the-art model checker)



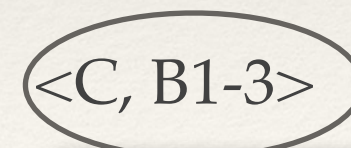
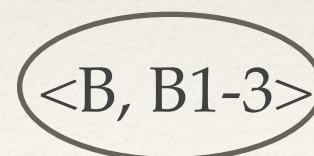
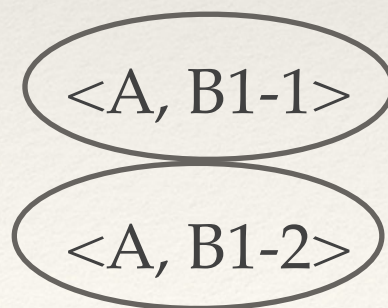
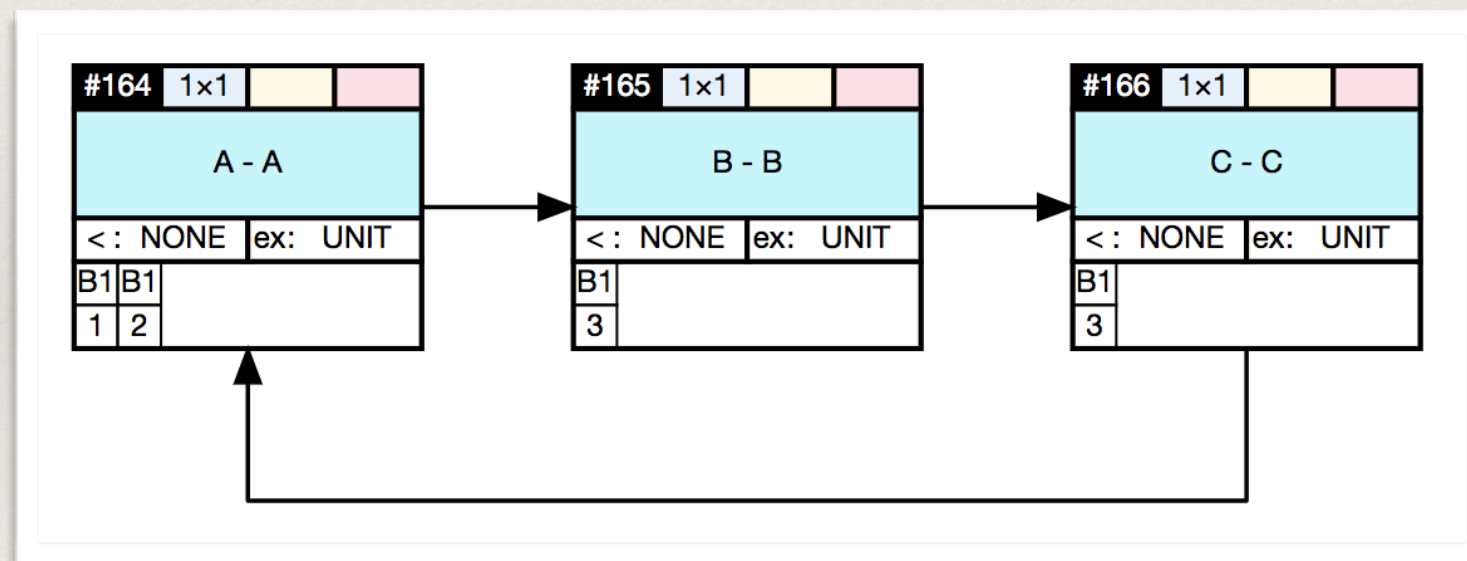
# How to Check Satisfiability?

	Model	Tasks	Dep.	Loc.	NuSMV
	Sat.	8	9	312	2min 35s
	Non-sat.	8	9	312	>1h



# Other Way to Check Satisfiability?

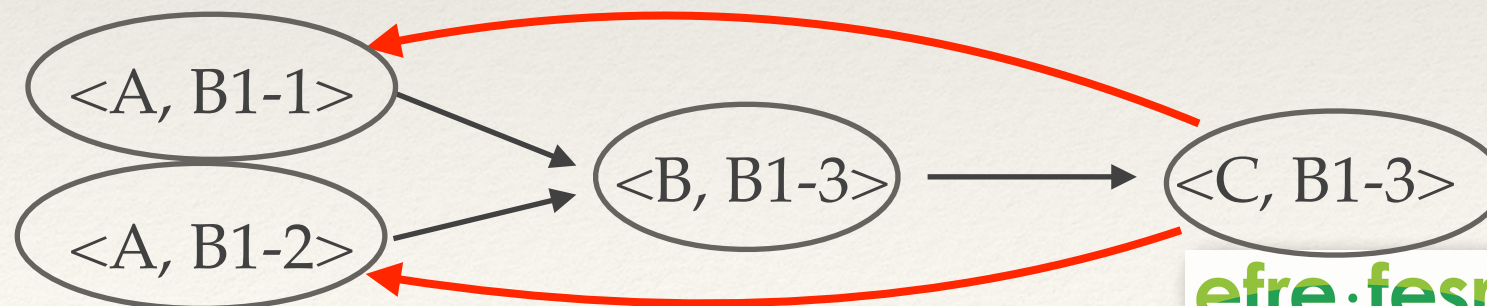
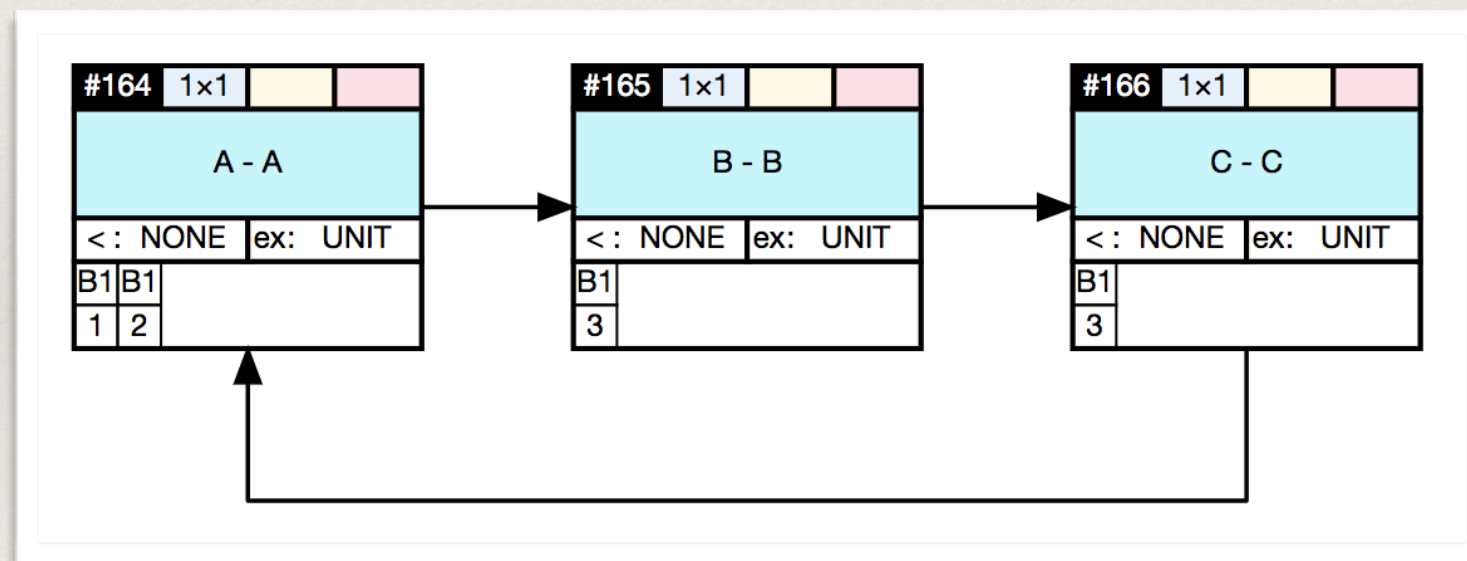
- Translate a Diagram into a Task-Unit (TU) Graph





# Other Way to Check Satisfiability?

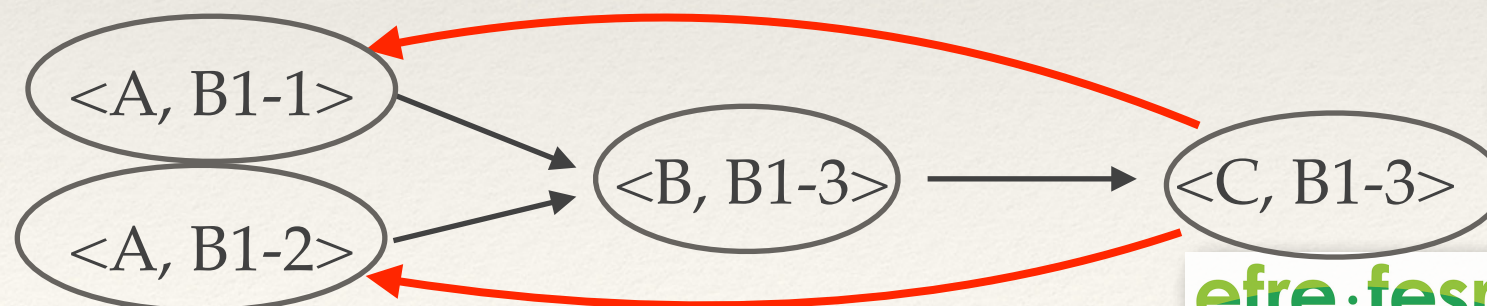
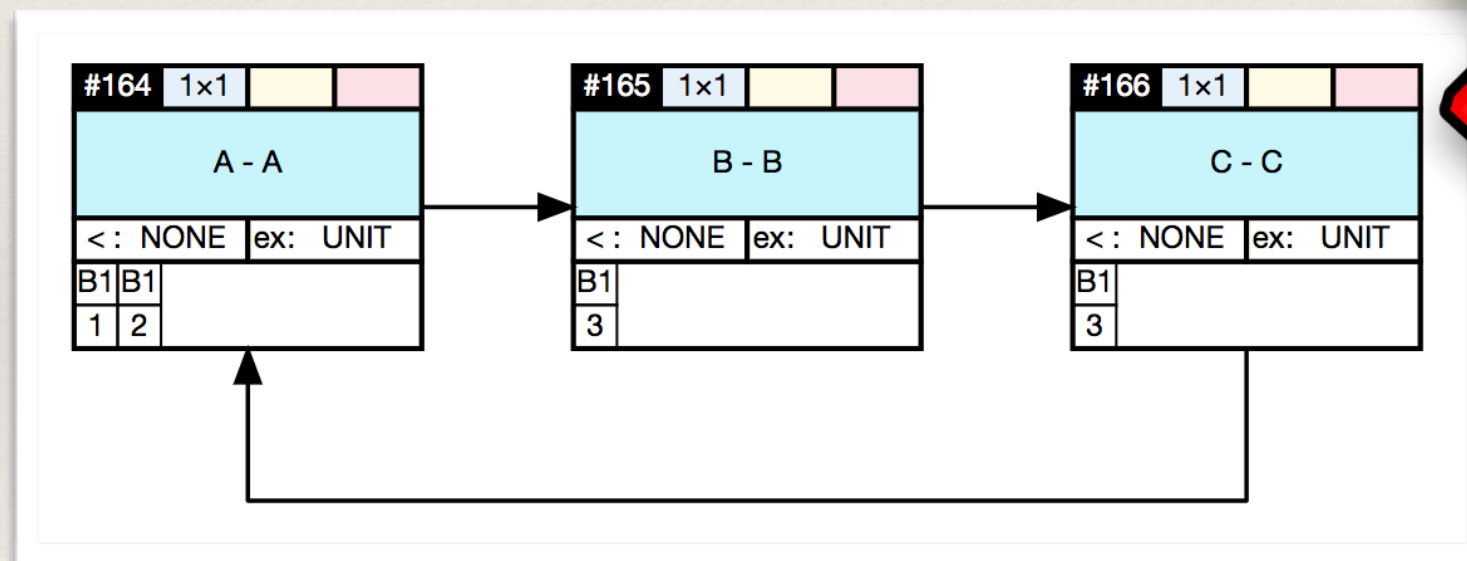
- Translate a Diagram into a Task-Unit (TU) Graph
- Translate the precedences into arrows between TU nodes





# Other Way to Check Satisfiability?

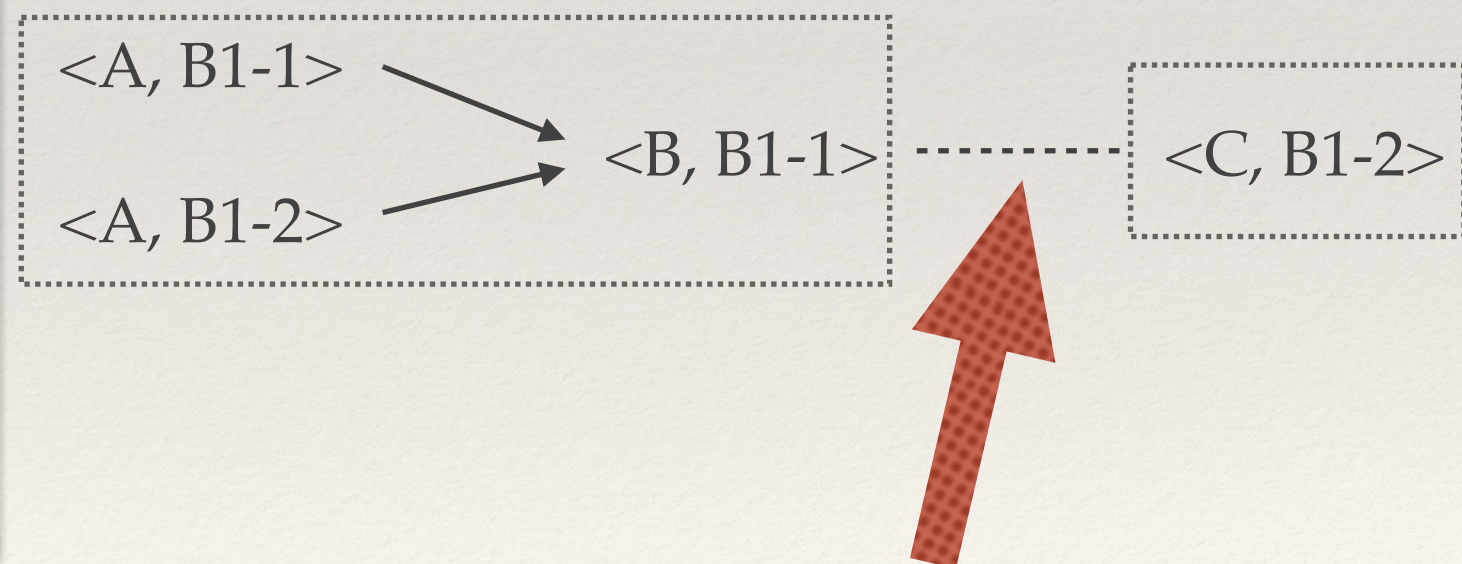
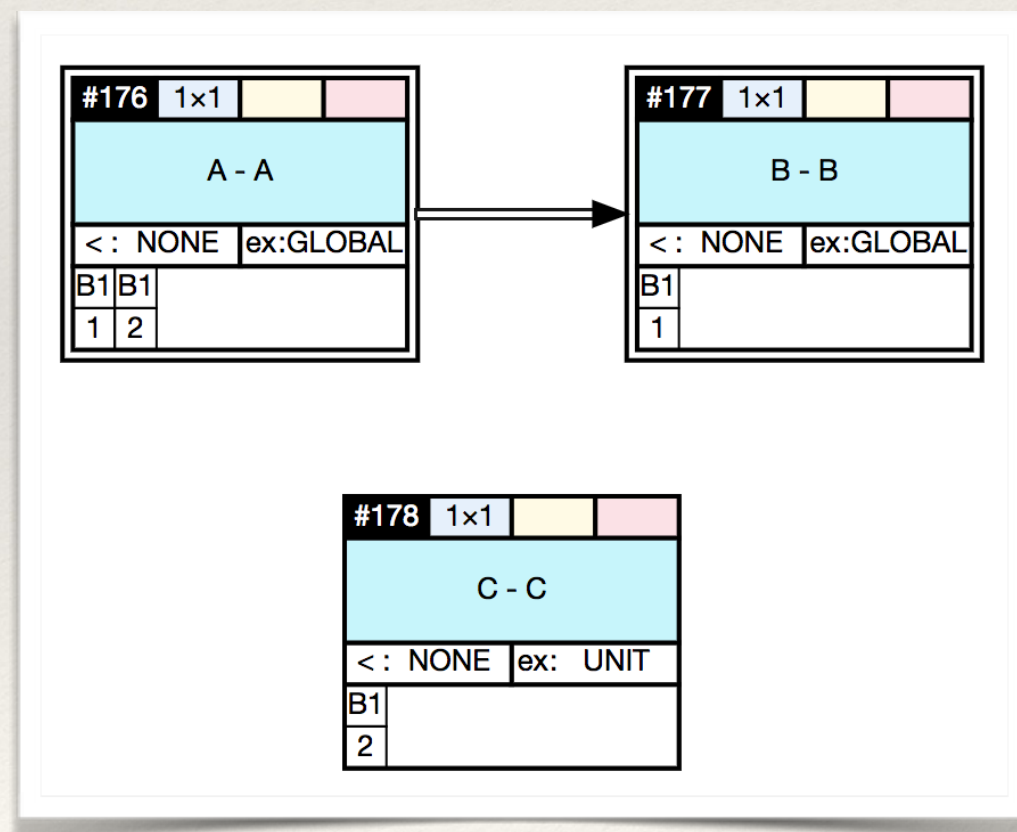
- Translate a Diagram into a Task-Unit (TU) Graph
- Translate the precedences into arrows between TU nodes
- Check for loops





# Disjunction in the TU Graph

- Some constraints introduce disjunction
- One has to check possible orientations





---

# Algorithm at a Glance

---

- **Check for Cycles**
  - Cycles: If the graph contains a cycle then is not orientable



# Algorithm at a Glance

- **Check for Cycles**
  - Cycles: If the graph contains a cycle then is not orientable
- **Deterministic Orientation**
  - Direct the undirected edges for which only one orientation is possible





# Algorithm at a Glance

- **Check for Cycles**
  - Cycles: If the graph contains a cycle then is not orientable
- **Deterministic Orientation**
  - Direct the undirected edges for which only one orientation is possible
- **Divide&Conquer**
  - Partition the graph so that:
    - orientability can be checked for each subgraph
    - by trying all orientations



# Satisfiability Check

	Model	Tasks	Dep.	Loc.	Nodes	Arcs	Edges	NuSMV	US
	Sat.	8	9	312	236	9415	524	2min 35s	27 ms
	Non-sat.	8	9	312	236	10003	521	>1h	5 ms



# Satisfiability Check

	Model	Tasks	Dep.	Loc.	Nodes	Arcs	Edges	NuSMV	US
✓	Sat.	8	9	312	236	9415	524	2min 35s	27 ms
✗	Non-sat.	8	9	312	236	10003	521	>1h	5 ms
✗	Bigger	12	14	312 (2)	244	9435	574	>1h	10 ms
✗	More Edges	12	14	312 (47)	424	15131	1740	>1h	23 ms



# Satisfiability Check

	Model	Tasks	Dep.	Nodes	Arcs	Edges	US
✓	Sat.	8	9	236	9415	524	27 ms
✗	Non-sat.	8	9	236	10003	521	5 ms
✗	Bigger	12	14	244	9435	574	10 ms
✗	More Edges	12	14	424	15131	1740	23 ms
✗	Bigger	480	1291	16,960	1,436,759	678,680	55,866 ms (~1 min)
✗	Bigger	720	2,526	25,440	3,082,925	1,526,820	379,409 ms (~6.32 min)
✗	Bigger	960	4,187	33,920	5,217,426	2,714,160	OOM



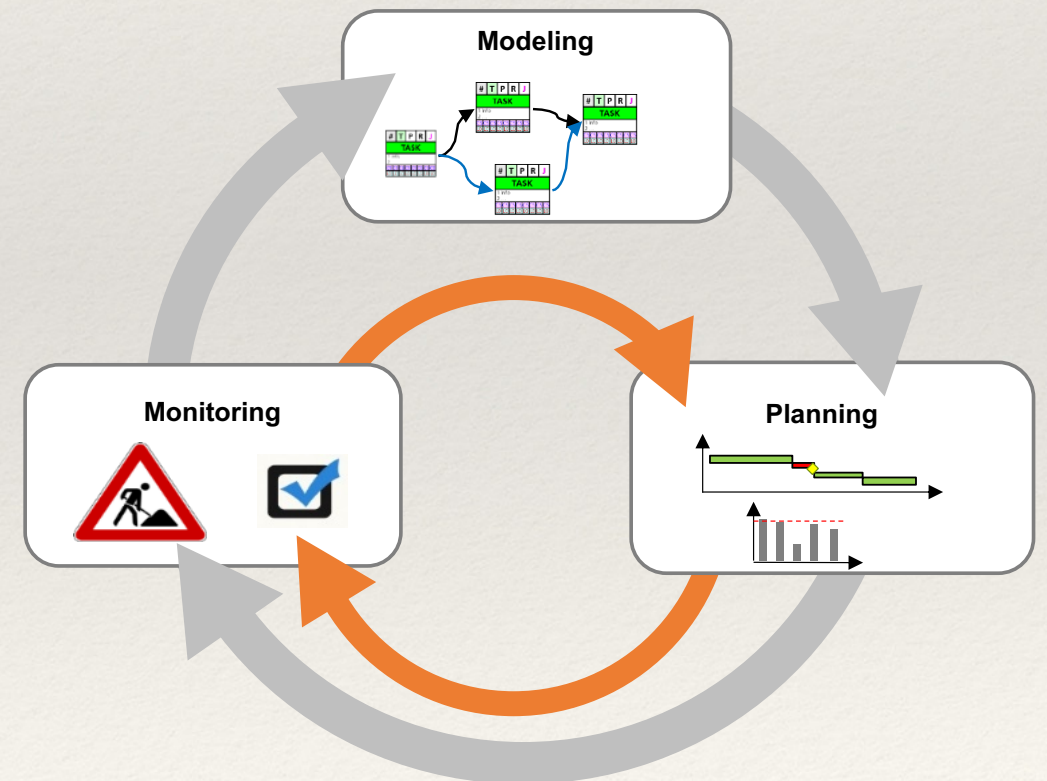
# Summary

- **CoPMod**: Constuction Process Modelling Language
  - Graphical
  - Declarative: captures **process requirements** (**what** and not **how**)
  - Formal
- Effective algorithm to check satisfiability
- proof-of-concept **tool** @ [copmod.inf.unibz.it](http://copmod.inf.unibz.it)

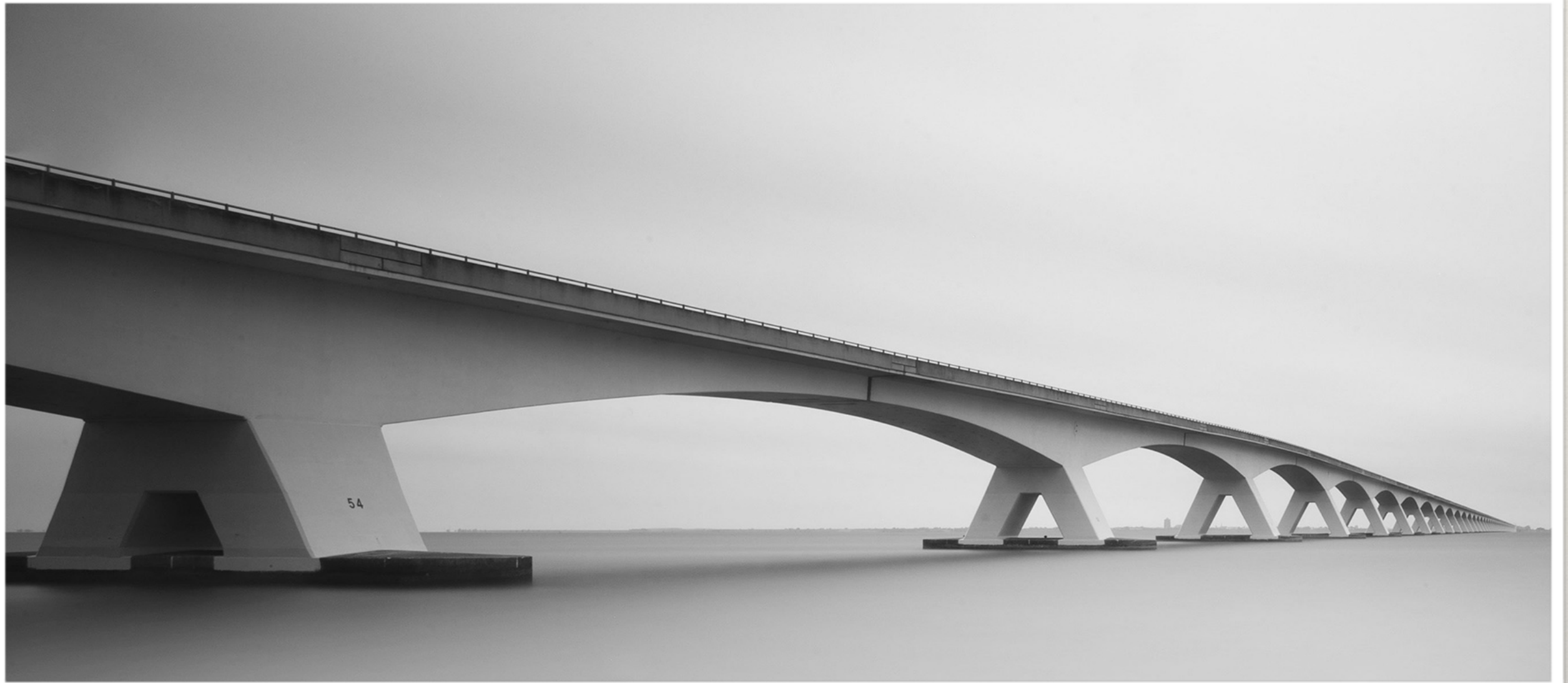


# Future Work

- COCKPiT:  
Collaborative Construction Project management
- Integrate Automatic Schedule:
  - Modeling
  - Automated Scheduling
  - Monitoring/Analysis







*KRDB — Summer Online Seminars*

---

**Thank you**

*Elisa Marengo*

Werner Nutt

Matthias Perktold

*Free University of Bolzano*