

---

## How to assign cost to “Avoidable Requirements Creep”

A step towards the waterfall’s agilization

Ralf Fahney & Dr. Stefan Puehl



## Motivation - The Paper in a Nutshell

- **The situation (sucks)**

- Complex fixed price projects deal inevitably with undetailed requirements.
- Experience and theory gap: rework is a real financial challenge triggered by the SRS.
- Risk ... planned for unavoidable creep only

- **Our suggested approach**

1. Construct each iteration with articulated change
2. Structure the plan of payments accordingly
3. Adjust the flow of payments
4. Plan the iteration scope with creep incorporated

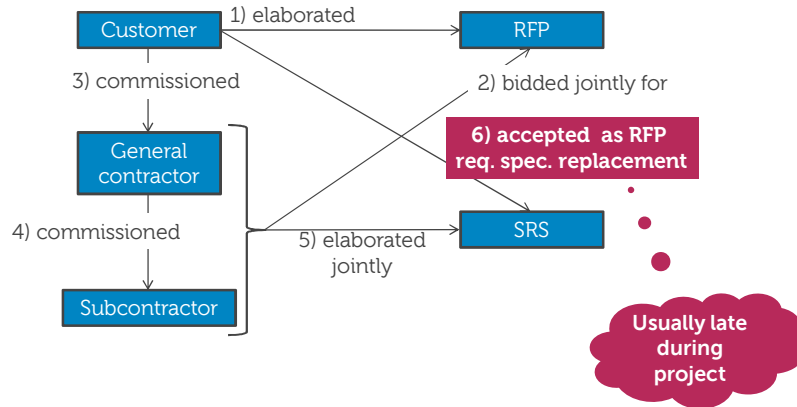
- **Your take-aways**

- Structure right from the start
- Address stability and flexibility at the same time
- Enable with the explicit requirement structure a robust customer dialogue on costs

- **Our benefits**

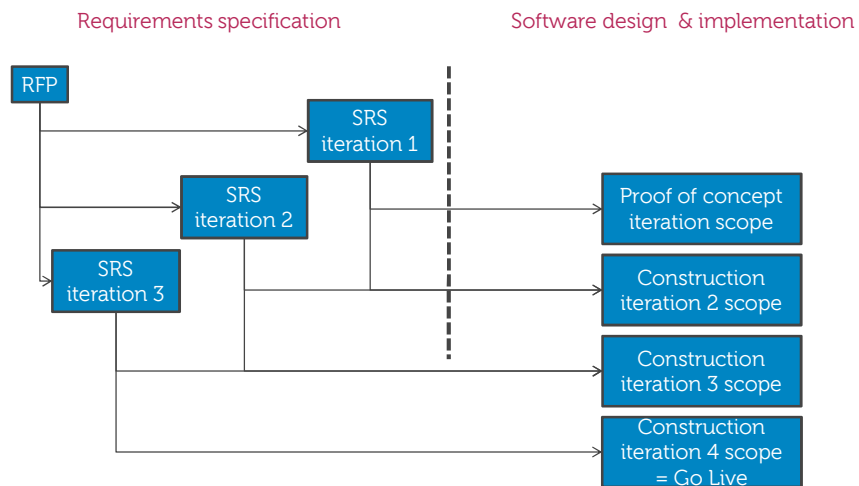
- Agile elements are defined as managed change for transparency.
- Waterfall –as fixed-price approach in global delivery– follows closer to customer learning and agreements.

## A sample/standard fixed price project setting



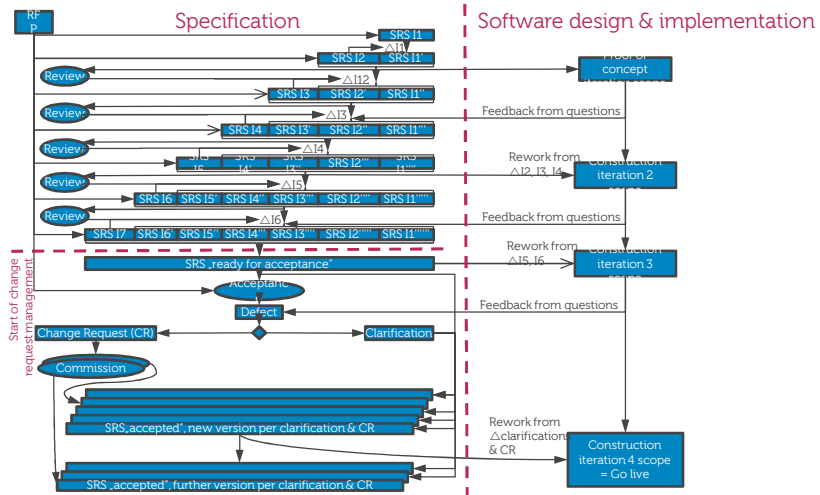
3

## How SRS elaboration is intended to work



4

## What really happens - Rework necessary but not planned



5

## The Problem – Our suggestion

How to organize a project to avoid "avoidable requirements creep" and thereby to reduce subsequent implementation rework?

**Stability**  
SRS acceptance step after each and every SRS iteration

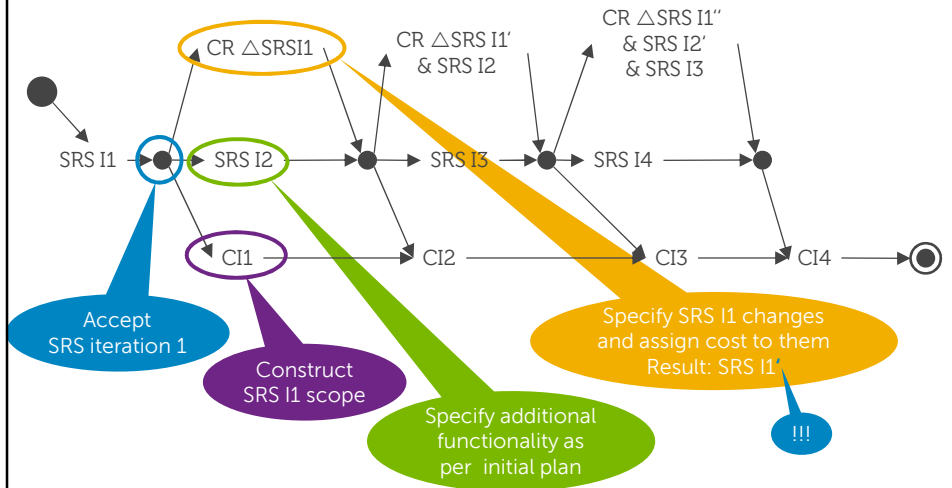


**Flexibility**  
Change request management at SRS level for each iteration



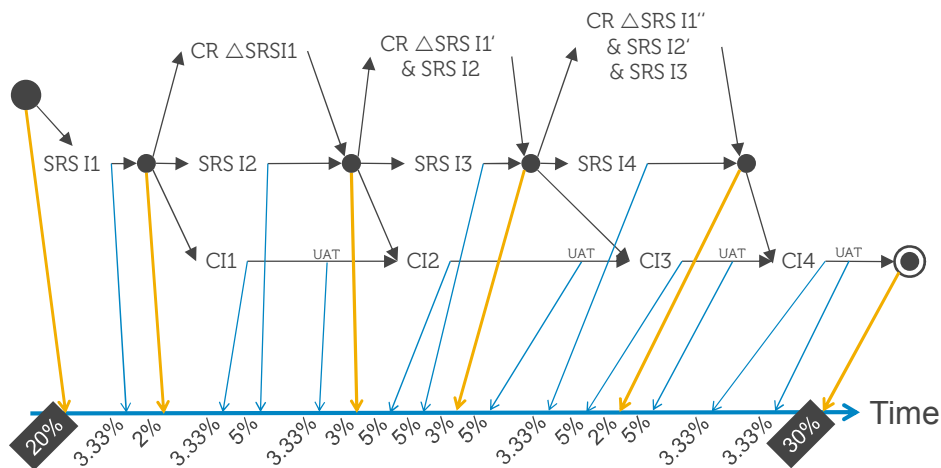
6

## #1 Construction iteration <n> scope



7

## #2 The actual flow of payments (ex. CRs) Event-driven, spread mostly equally over time



8

### #3 The initial plan of payments

Scope planned to be increased by x%		20%	30%	30%	20%
Total of scope achieved at the iteration end		20%	50%	80%	100%
Effort spent for	Payment trigger	Amount			
Requirements (1/3)		33%	SRS I1	SRS I2	SRS I3 SRS I4
	Contract signment	20%	1.33%	2.00%	2.00% 1.33%
	SRS delivery	50%	3.33%	5.00%	5.00% 3.33%
	RFP req. section replacement acceptance	30%	2.00%	3.00%	3.00% 2.00%
Construction (2/3)		67%	CI1	CI2	CI3 CI4
	Contract signment	20%	2.67%	4.00%	4.00% 2.67%
	Implementation delivery	25%	3.33%	5.00%	5.00% 3.33%
	User acceptance test passed	25%	3.33%	5.00%	5.00% 3.33%
	IT system productive deployment	30%	4.00%	6.00%	6.00% 4.00%
Change requests might apply to a maximum of x% of scope			0%	20%	50% 80%

Payments are updated according to CRs from rework.

9

### Benefits come with challenges

#### Customer Benefits

- Joint proceeding on safe ground
- Negotiations take place as early as possible
- Flow of payments supports to initiate negotiations.
- Flow of payments motivates supplier to work coordinated
- Mutual confidence to build the right system
- Any type of requirements creep processed uniformly

#### Challenges

- As supplier you're called nitpicker, killjoy, spoilsport
- Customer might not like to take responsibility
- Plan of payments might be difficult to explain to customer
- Procedure maybe not perceived as fixed price  
(even though still NOT "time & material")
- Challenge to identify core and self-contained functionality is not addressed by cost only
- Maybe difficult to convince that creep will occur anyway

10

## Conclusion

- Apply Change Request Management to SRS artifacts in the fixed price frame of reference and work for incremental acceptance
- Watch out , plan and raise discussion as early as possible
  - Seems to be counter-intuitive because everything seems to be defined.
  - But: Customers appreciate to leverage iteration scope transparency.
- Our progress:
  - CR are never discussed in the beginning of a project; RE can change that.
  - Step to converge waterfall and agile in a global service delivery setting
  - But: further domain questions remain

11

# Thank you

stefan\_puehl@dell.com  
rf@fahney.com

12