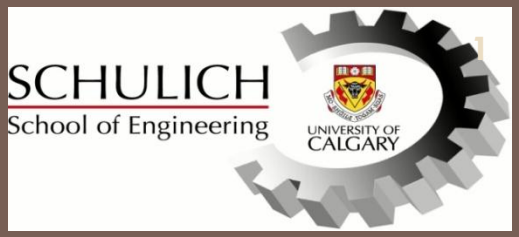


THE HOW? WHEN? AND WHAT? FOR THE PROCESS OF RE-PLANNING FOR PRODUCT RELEASES

Anas Jadallah[†], Ahmed Al-Emran[†], Mahmood Moussavi[†], Guenther Ruhe[†*&]

[†]*Department of Electrical & Computer Engineering, University of Calgary*

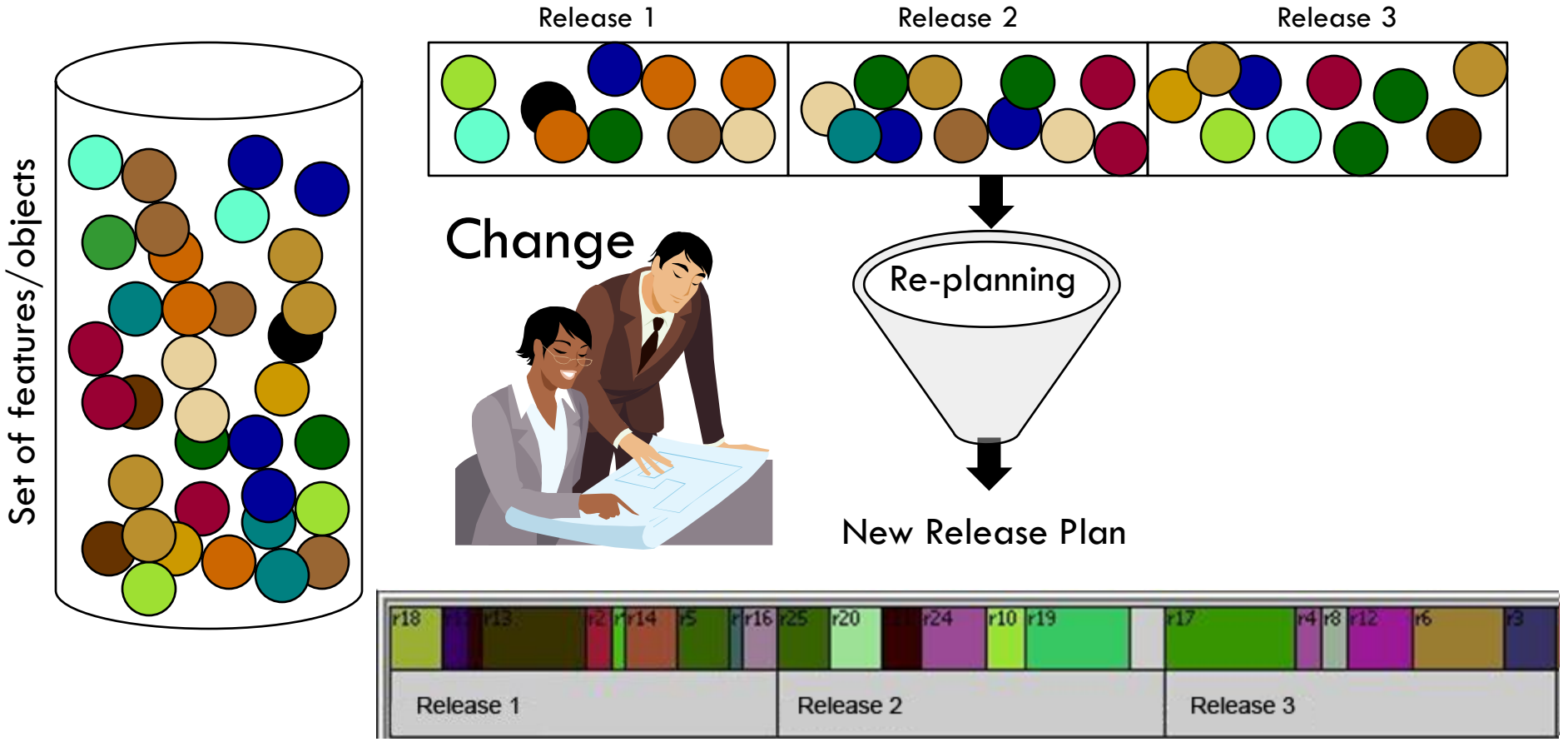
[*]*Department of Computer Science, University of Calgary*

[&]*Expert Decisions Inc, Calgary*

May, 2009

Introduction

- Release planning assigns features to a set of releases to address various constraints.
- Requirements are volatile and changing frequently during software development.
- Modifications to the baseline Release plan is needed to accommodate these changes.

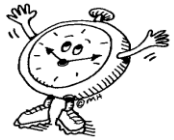

Release 1
Release 2
Release 3
Set of features/objects
Change
Re-planning
New Release Plan
r18
r13
r2
r14
r5
r16
r25
r20
r24
r10
r19
r17
r4
r8
r12
r6
r3
Release 1
Release 2
Release 3

Motivation

- Changing requirements is considered as one of the major causes for software product failure[1].
- A well established process for handling re-planning of product releases is strongly needed [2,3,4].

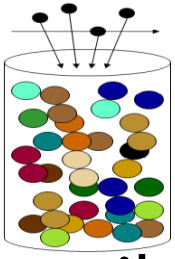
Research Problem

- Release period

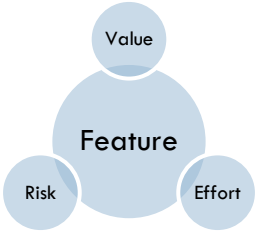


- Release effort capacity

- Set of features

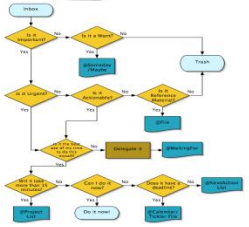


- Set of change requests
- Each feature has its own attributes



- Time to start re-planning is not known

- A well defined process to select best features is needed.

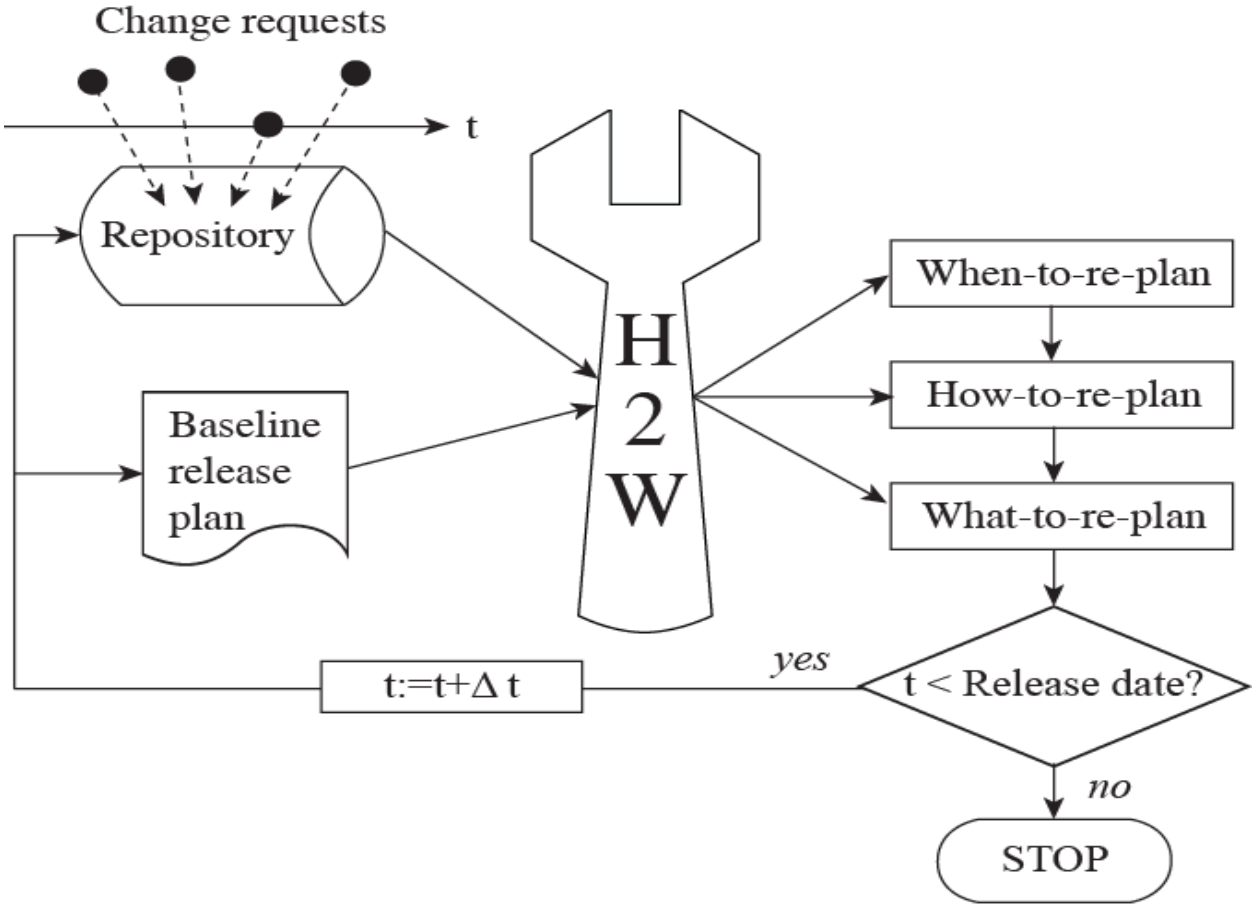


- There is allowed degree of change

Research Focus

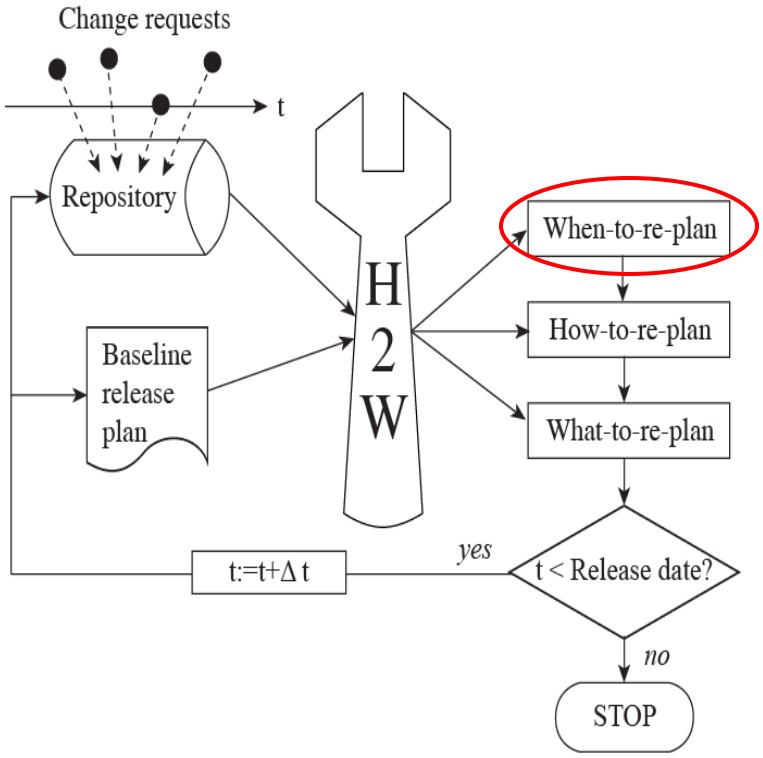
- **WHEN** shall we start the re-planning process (What triggers the re-planning) ?
- **HOW** can we decide which features or change requests are most attractive to incorporate?
- **WHAT** degree of change in the already announced release plan is acceptable (How to compromise between adding more attractive features and maintaining the release stability) ?

Research Method (H2W)



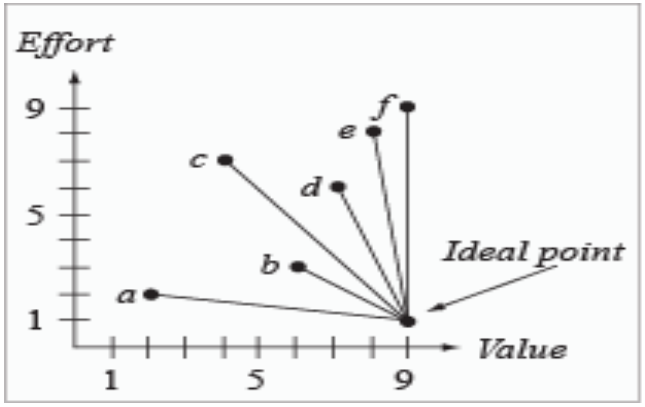
Step 1: When to Re-plan

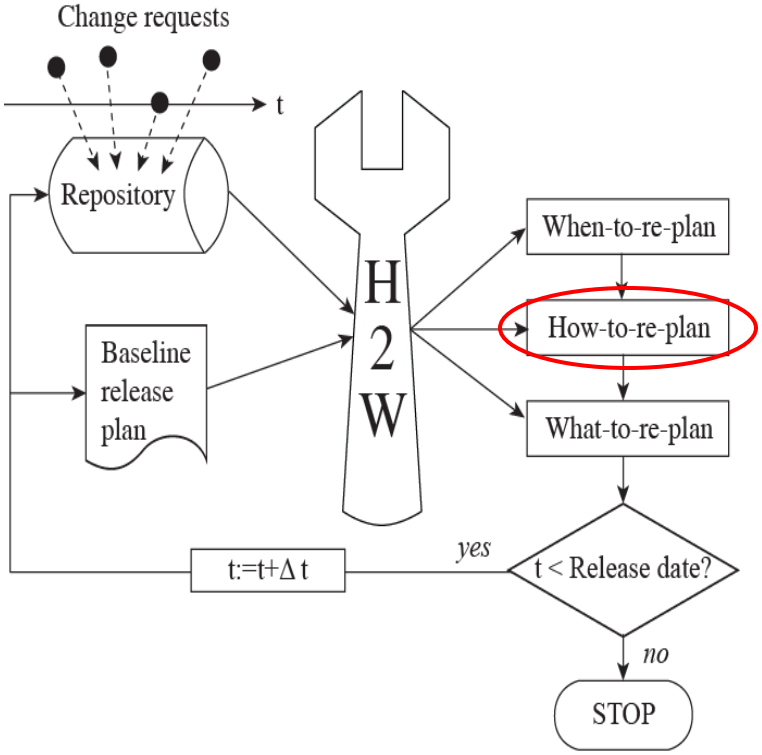
- This decision is based on a threshold related to the accumulation of all arriving features:
  - Number of accumulated features
  - Accumulated value of these features
  - Defect slippage rate
- We consider a value based threshold (V-THRESHOLD)



Step 2: How to Re-plan

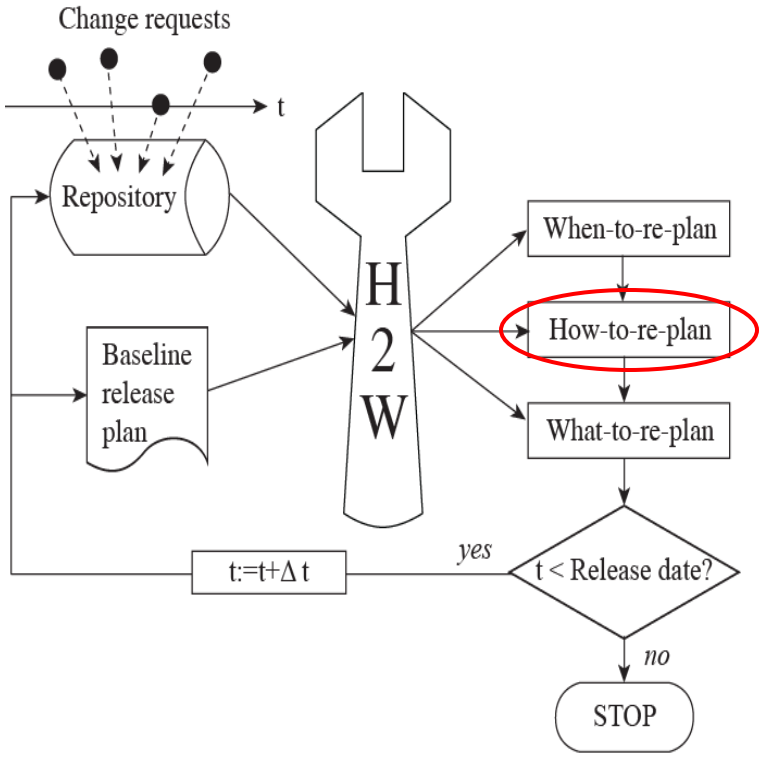
- Each feature is mapped in to a point in a three dimension space.
- Features' attributes are assumed to be as a result of experts' evaluation
- Ideal point is defined as the best artificial solution having
  - extremely high value
  - extremely low effort
  - extremely low risk





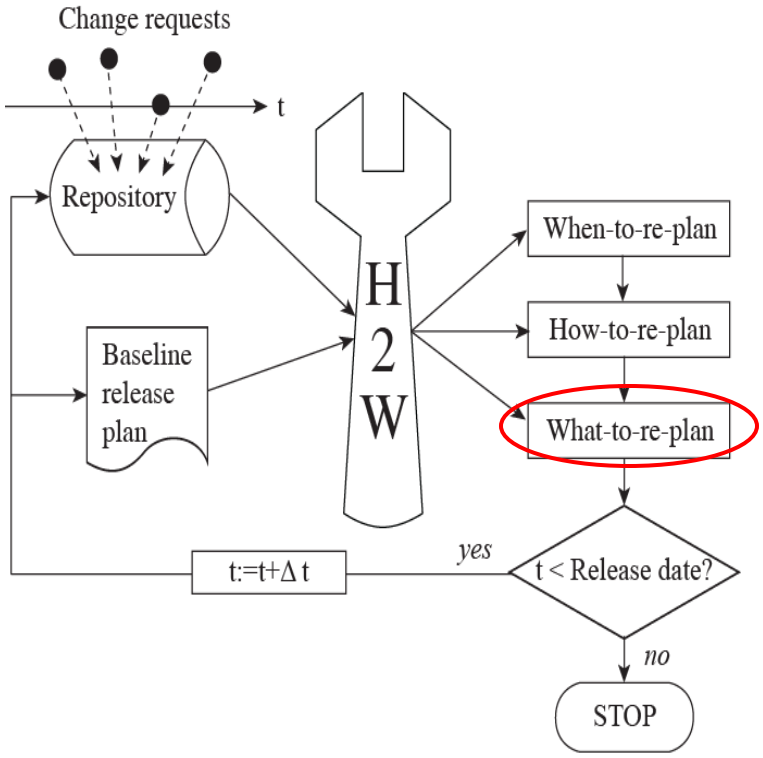
Step 2: How to Re-plan

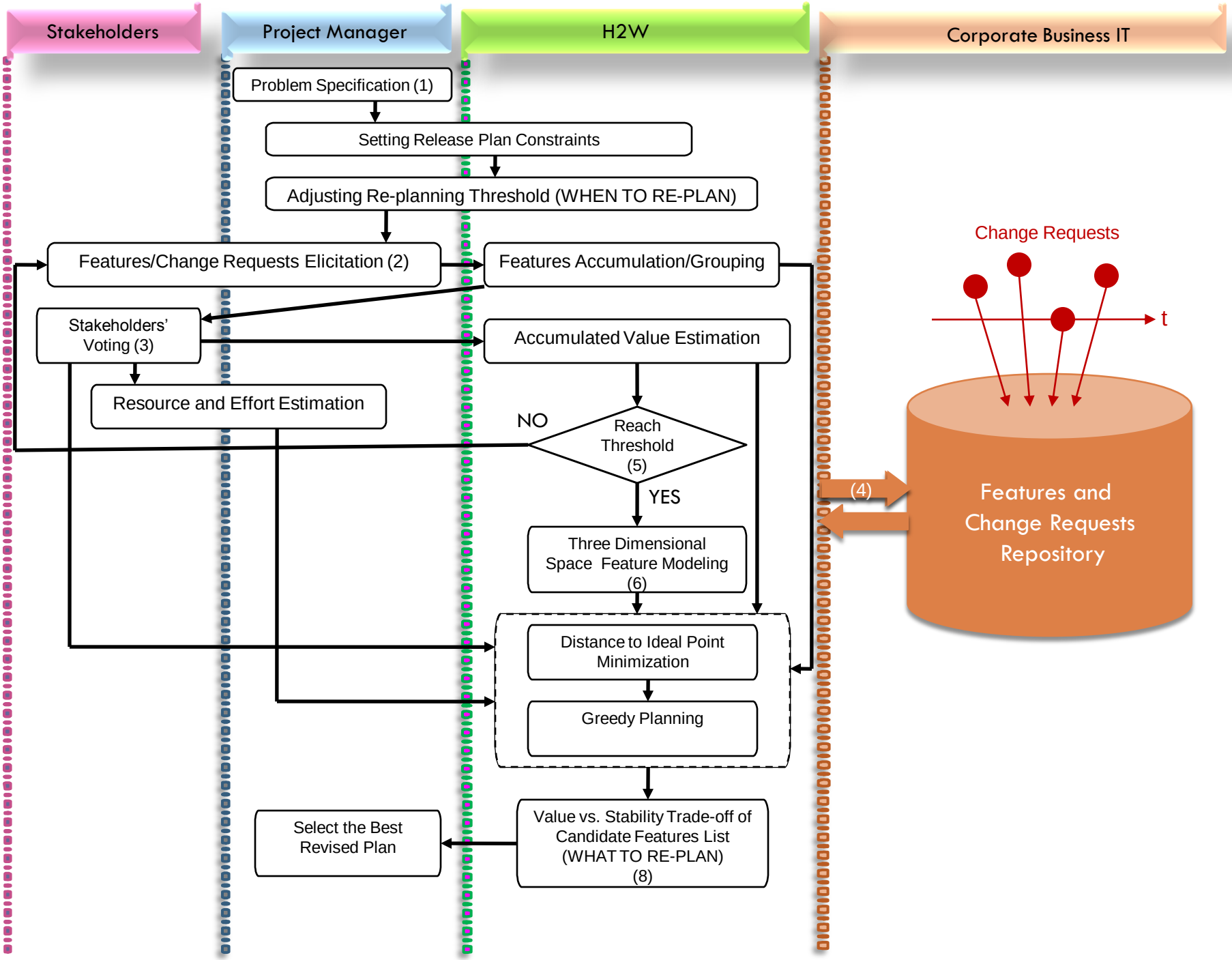
- The Euclidian distance is measured between each feature and the ideal point.
- The goal is to come as close as possible to the ideal solution
- Candidate features are ranked based on their distance to the ideal point.
- Greedy optimization is applied.



Step 3: What to Re-plan

- The normalized increase in value is plotted against the normalized decrease in stability of the plan.
- Stability is defined in terms of the number of features changed relative to the total number of possible modifications.
- The point of intersection determines the best number of features to exchange.



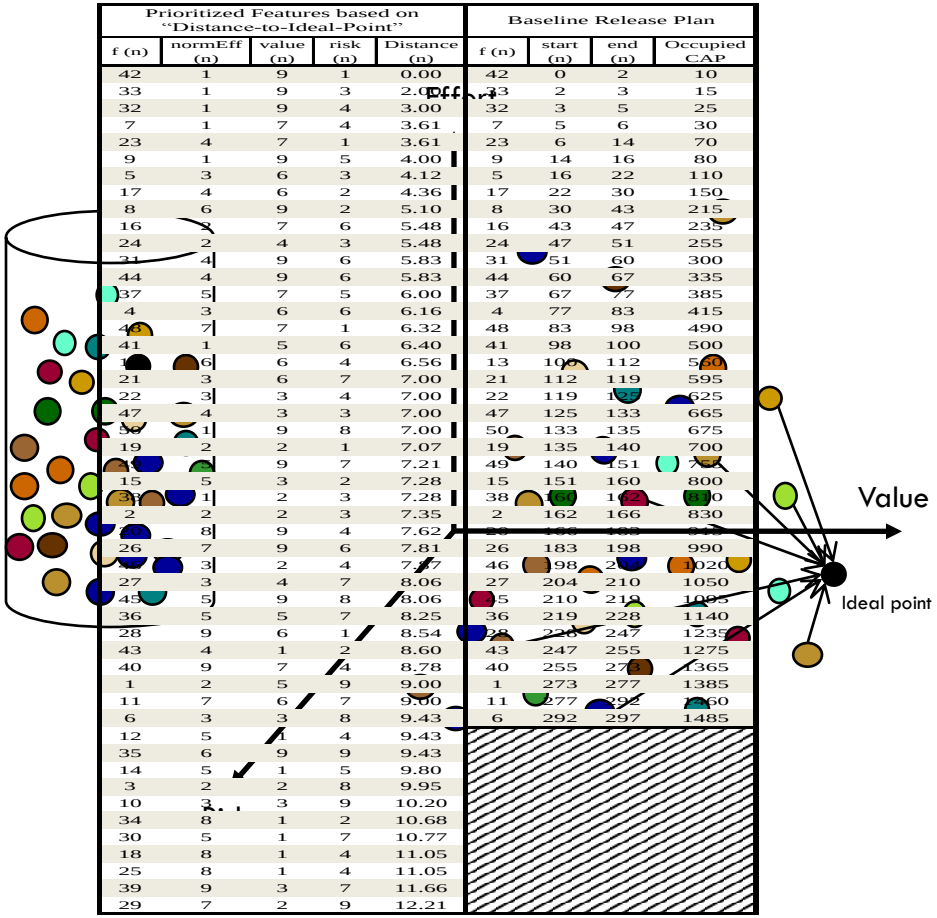

Stakeholders
Project Manager
H2W
Corporate Business IT
Problem Specification (1)
Setting Release Plan Constraints
Adjusting Re-planning Threshold (WHEN TO RE-PLAN)
Features/Change Requests Elicitation (2)
Features Accumulation/Grouping
Stakeholders' Voting (3)
Accumulated Value Estimation
Resource and Effort Estimation
NO
Reach Threshold (5)
YES
Three Dimensional Space Feature Modeling (6)
Distance to Ideal Point Minimization
Greedy Planning
Value vs. Stability Trade-off of Candidate Features List (WHAT TO RE-PLAN) (8)
Select the Best Revised Plan
(4)
Change Requests
t
Features and Change Requests Repository

Case Study

- The studied problem is based on real world data for planning a project for Expert Decision Inc.
- Project: Decision support tool ReleasePlanner®
- It involves the following initial problem parameters:
  - Set of fifty features;
  - Release start time, T1 = 0;
  - Release end time, T2 = 300;
  - Effort capacity, CAP = 1500.

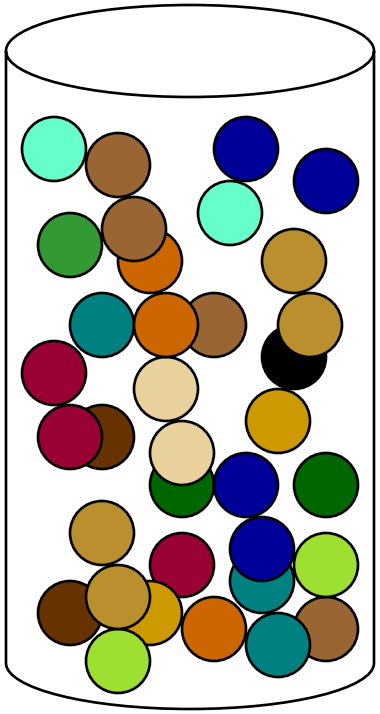
Baseline planning

- Each Feature is mapped in to a point in three dimensional space.
- Distance-to-Ideal-Point minimization was applied to rank features.
- Greedy planning was used to select features.
- This results in 39 features to be accommodated in the next release.
- A total of 11 features are rejected from the baseline plan.
- A release value of 236 can be achieved.



STEP 1: When to Re-plan

- Value based threshold triggers re-planning.
- V-THRESHOLD value is 25% of the baseline release plan.
- 20 change request were stochastically generated to simulate real world situation.



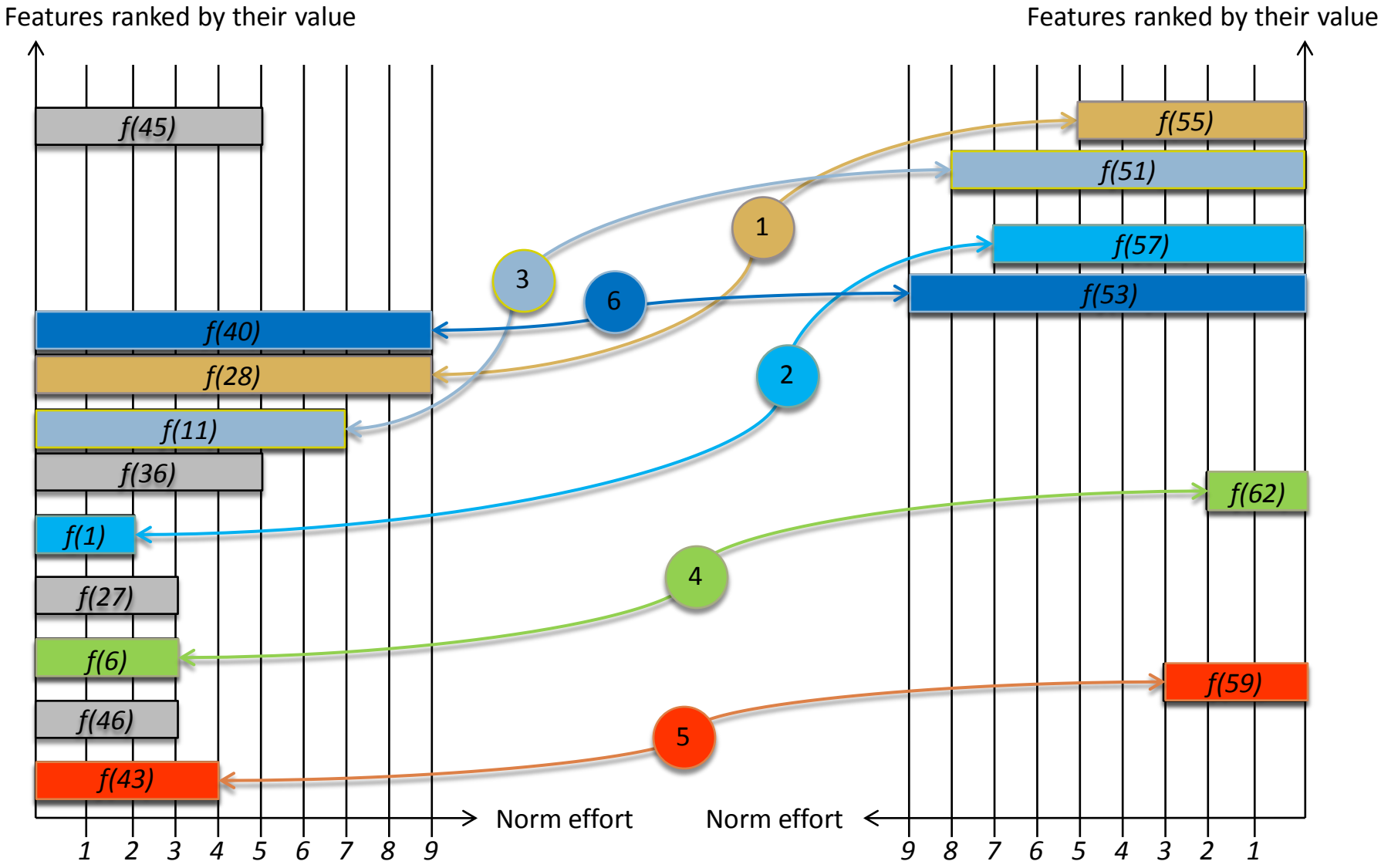
Change Requests					
f (n)	effort (n)	value (n)	risk (n)	time (n)	Cumulative value (n)
51	83	9	3	10	9
52	65	2	4	11	11
53	96	8	3	42	19
54	96	5	8	43	24
55	45	9	8	68	33
56	67	2	8	77	35
57	70	8	7	105	43
58	67	3	7	118	46
59	31	2	2	142	48
60	71	4	9	150	52
61	47	5	8	165	57
62	22	5	5	194	62
63	73	8	4	204	70
64	67	3	2	210	73
65	97	4	6	239	77
66	77	4	6	248	81
67	90	9	6	275	90
68	19	4	8	281	94
69	83	3	9	288	97
70	23	4	6	292	101

STEP 2: How to Re-plan

At the time of re-planning:

- 29 out of 39 features were implemented.
- The remaining effort Capacity was 510
- Rank candidate features and change requests by their distance to ideal point.
- Greedy optimization is applied again.

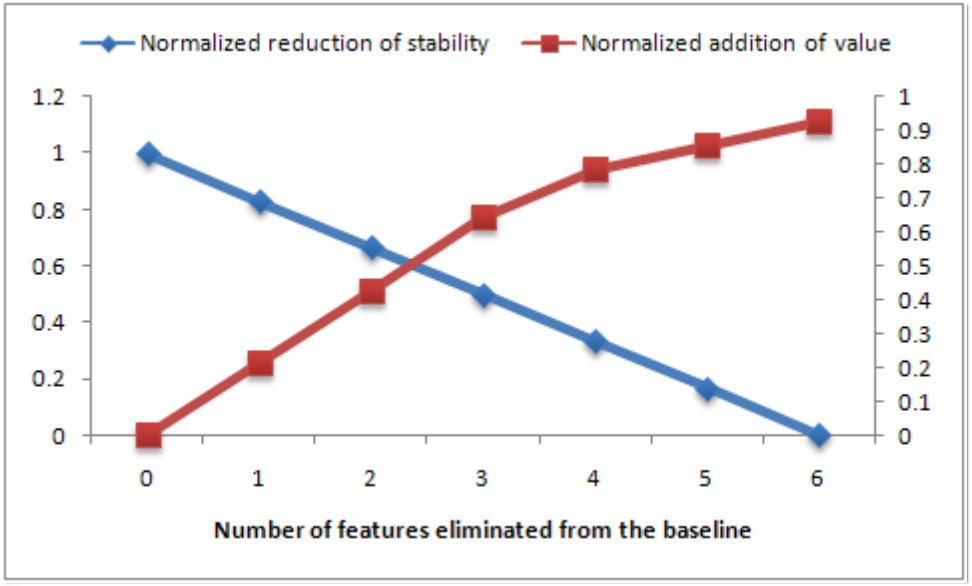
Prioritized Candidates at Re-planning Point Based on "Distance-to-Ideal-Point"					Prioritized Candidates at Re-planning Point Based on "Distance-to-Ideal-Point"				
f (n)	normEff (n)	value (n)	risk (n)	Distance (n)	f (n)	normEff (n)	value (n)	risk (n)	Distance (n)
62	2	5	5	5.74	6	3	3	8	9.43
51	8	9	3	7.28	12	5	1	4	9.43
59	3	2	2	7.35	35	6	9	9	9.43
46	3	2	4	7.87	14	5	1	5	9.80
27	3	4	7	8.06	3	2	2	8	9.95
45	5	9	8	8.06	10	3	3	9	10.20
55	5	9	8	8.06	58	7	3	7	10.39
36	5	5	7	8.25	34	8	1	2	10.68
53	9	8	3	8.31	30	5	1	7	10.77
28	9	6	1	8.54	18	8	1	4	11.05
57	7	8	7	8.54	25	8	1	4	11.05
43	4	1	2	8.60	60	7	4	9	11.18
40	9	7	4	8.78	54	9	5	8	11.36
1	2	5	9	9.00	56	7	2	8	11.58
11	7	6	7	9.00	39	9	3	7	11.66
61	5	5	8	9.00	29	7	2	9	12.21
52	6	2	4	9.11					


Features ranked by their value
f(45)
f(40)
f(28)
f(11)
f(36)
f(1)
f(27)
f(6)
f(46)
f(43)
1 2 3 4 5 6 7 8 9
Norm effort
Features ranked by their value
f(55)
f(51)
f(57)
f(53)
f(62)
f(59)
9 8 7 6 5 4 3 2 1
Norm effort
1
2
3
4
5
6

STEP 3: What to Re-plan

- Compromising the added value with the degree of Change

Number of features eliminated from baseline	Set of eliminated features	Set of replacement features	Added value
1	{28}	{55}	3
2	{28,1}	{55,57}	3+3
3	{28,1,11}	{55,57,51}	3+3+3
4	{28,1,11,6}	{55,57,51,62}	3+3+3+2
5	{28,1,11,6,43}	{55,57,51,62,59}	3+3+3+2+1
6	{28,1,11,6,43,40}	{55,57,51,62,59,53}	3+3+3+2+1+1



Conclusion

- Re-planning is important, otherwise significant information is not taken into account
- The usefulness of this approach is that it:
  - Provides a comprehensive method for re-planning.
  - Allows easy and quick re-planning for problems of at least 50 + 20 features.
  - Compromises between common issues in re-planning (value-based plans)and practical managers concerns (changing announced plan)
- Limitations of the internal validity of the results :
  - Simplified model for representing the effort.
  - Some form of operational planning needs to be in place to track features status at re-planning time.
  - The greedy optimization may not necessarily produce optimal results

Future Work

- Considering dependencies between features.
- Creating additional functionality Vs. stabilizing the existing code and fixing detected defects.
- Considering other types of re-planning thresholds.
- Investigate the sensitivity of solutions depending on determined threshold value.
- Industrial evaluation of the H2W method.
- Tool support for H2W method.

References

1. Van Lamsweerde, A., "Requirements Engineering in the Year 00: a research perspective", Proceedings of the 22nd International Conference on Software Engineering (ICSE' 2000), Limerick, Ireland, 2000.
2. Stark, G., Skillicorn, A. & Ameele, R., “An Examination of the Effects of Requirements Changes on Software Maintenance Releases”, Journal of Software Maintenance: Research and Practice, 1999.
3. AlBourae, T., Ruhe, G. & Moussavi, M., “Lightweight Re-planning of Software Product Releases”, 14th IEEE International Requirements Engineering Conference Minneapolis/St. Paul, Minnesota, USA, 2006.
4. Loconsole, A., "Empirical Studies on Requirement Management Measures", in Proceedings - 26th International Conference on Software Engineering, ICSE 2004, Edinburgh, United Kingdom, 2004.