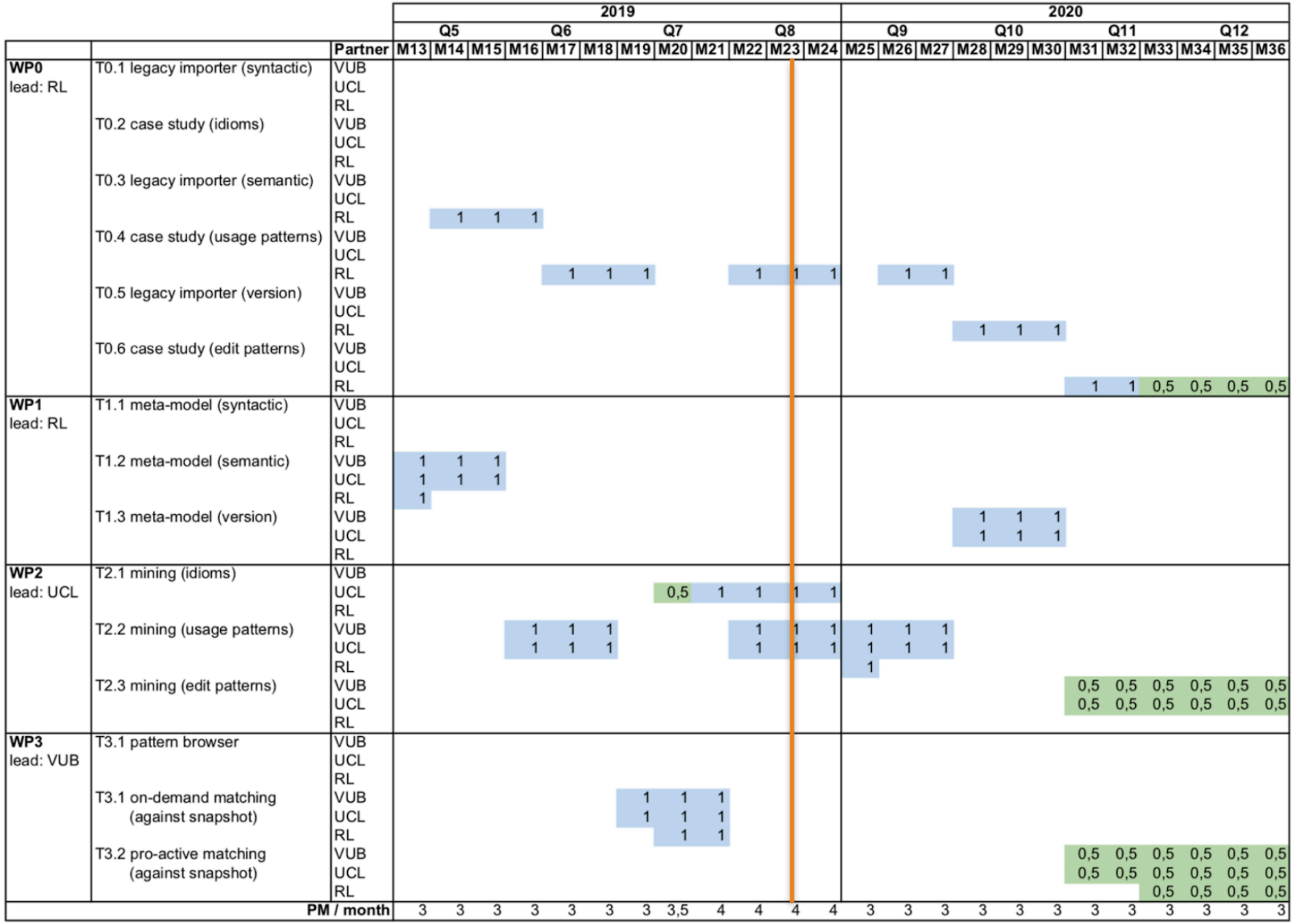

			2019												2020											
			Q5			Q6			Q7			Q8			Q9			Q10			Q11			Q12		
		Partner	M13	M14	M15	M16	M17	M18	M19	M20	M21	M22	M23	M24	M25	M26	M27	M28	M29	M30	M31	M32	M33	M34	M35	M36
WP0 lead: RL	T0.1 legacy importer (syntactic)	VUB																								
		UCL																								
		RL																								
	T0.2 case study (idioms)	VUB																								
		UCL																								
		RL																								
	T0.3 legacy importer (semantic)	VUB																								
		UCL																								
		RL		1	1	1																				
	T0.4 case study (usage patterns)	VUB																								
		UCL																								
		RL					1	1	1			1	1	1		1	1									
	T0.5 legacy importer (version)	VUB																								
		UCL																								
		RL																1	1	1						
	T0.6 case study (edit patterns)	VUB																								
		UCL																								
		RL																			1	1	0,5	0,5	0,5	0,5
WP1 lead: RL	T1.1 meta-model (syntactic)	VUB																								
		UCL																								
		RL																								
	T1.2 meta-model (semantic)	VUB	1	1	1																					
		UCL	1	1	1																					
		RL	1																							
	T1.3 meta-model (version)	VUB																1	1	1						
		UCL																1	1	1						
		RL																								
WP2 lead: UCL	T2.1 mining (idioms)	VUB																								
		UCL								0,5	1	1	1	1												
		RL																								
	T2.2 mining (usage patterns)	VUB				1	1	1				1	1	1	1	1	1									
		UCL				1	1	1				1	1	1	1	1	1									
		RL													1											
	T2.3 mining (edit patterns)	VUB																			0,5	0,5	0,5	0,5	0,5	0,5
		UCL																			0,5	0,5	0,5	0,5	0,5	0,5
		RL																								
WP3 lead: VUB	T3.1 pattern browser	VUB																								
		UCL																								
		RL																								
	T3.1 on-demand matching (against snapshot)	VUB							1	1	1															
		UCL							1	1	1															
		RL								1	1															
	T3.2 pro-active matching (against snapshot)	VUB																			0,5	0,5	0,5	0,5	0,5	0,5
		UCL																			0,5	0,5	0,5	0,5	0,5	0,5
		RL																					0,5	0,5	0,5	0,5
		PM / month	3	3	3	3	3	3	3	3,5	4	4	4	4	3	3	3	3	3	3	3	3	3	3	3	3

Deliverables - Y2

Report to Innoviris due by end of March

- P0.3 Semantic extension of legacy importer (M16)

needs to be extended with the GROUM part

- R0.4 Evaluation of support for mining usage patterns (M24)

COEN reminds TIM

- P1.3 Semantic metamodel (M15)

Tim says that the Cobol GROUM extractor is close to finished, needs to be applied to BartholBank example in order to write up deliverable, ideally we also try to apply it to larger projects

- P1.4 Its implementation (M15)
- R2.2b Extra Freq-T evaluation (M24)

Celine's work on syntactic patterns - cloning BENEVOL paper + Johan's improvements to the algorithm

- P2.3 Miner semantic patterns (M18)

Tim: point to correct readme's.

- R2.4 Evaluation of miner (M24)

BigGroum applied to Java projects using Swing;

- P3.2 Browser for partial matches (M21)

partial matcher for syntactic patterns has been implemented, deliverable is almost written opportunity for Johan to integrate in Pharo tool

Tim: example output

Deliverables - Y3

- *Next progress report: end of M27 (March '20)*
- P0.5 Version information extension of legacy importer (M31) RL (with VUB help)
- R0.6 Evaluation of support for edit patterns (M36) RL (with VUB help) evaluation of change pattern miner on C#
- P1.5 Change-centric metamodel (M30)
- P1.6 Its implementation (M31)
- P2.3b Subdue algorithm (M27) UCL: subdue for usage patterns (rather than BigGroum) implementation done; report is being written
- P2.4b Use case Subdue-Biggroum comparison (M27) UCL: evaluation (turns out to be slower than BigGroum)
- P2.5 Miner edit patterns (M36) VUB: miner for change patterns (Facebook's GetAFix vs VUB's SysEdMiner) UCL: difference between structural patterns found in separate versions
- R2.6 Evaluation of miner (M36) battle of the miners :)
- P3.3 Edit recommender to complete partial match of a mined pattern (M36)

refocus to partial matches, possibly easy in Pharo tool
or do more on BigGroum for Cobol?
NOT argue for this at progress meeting