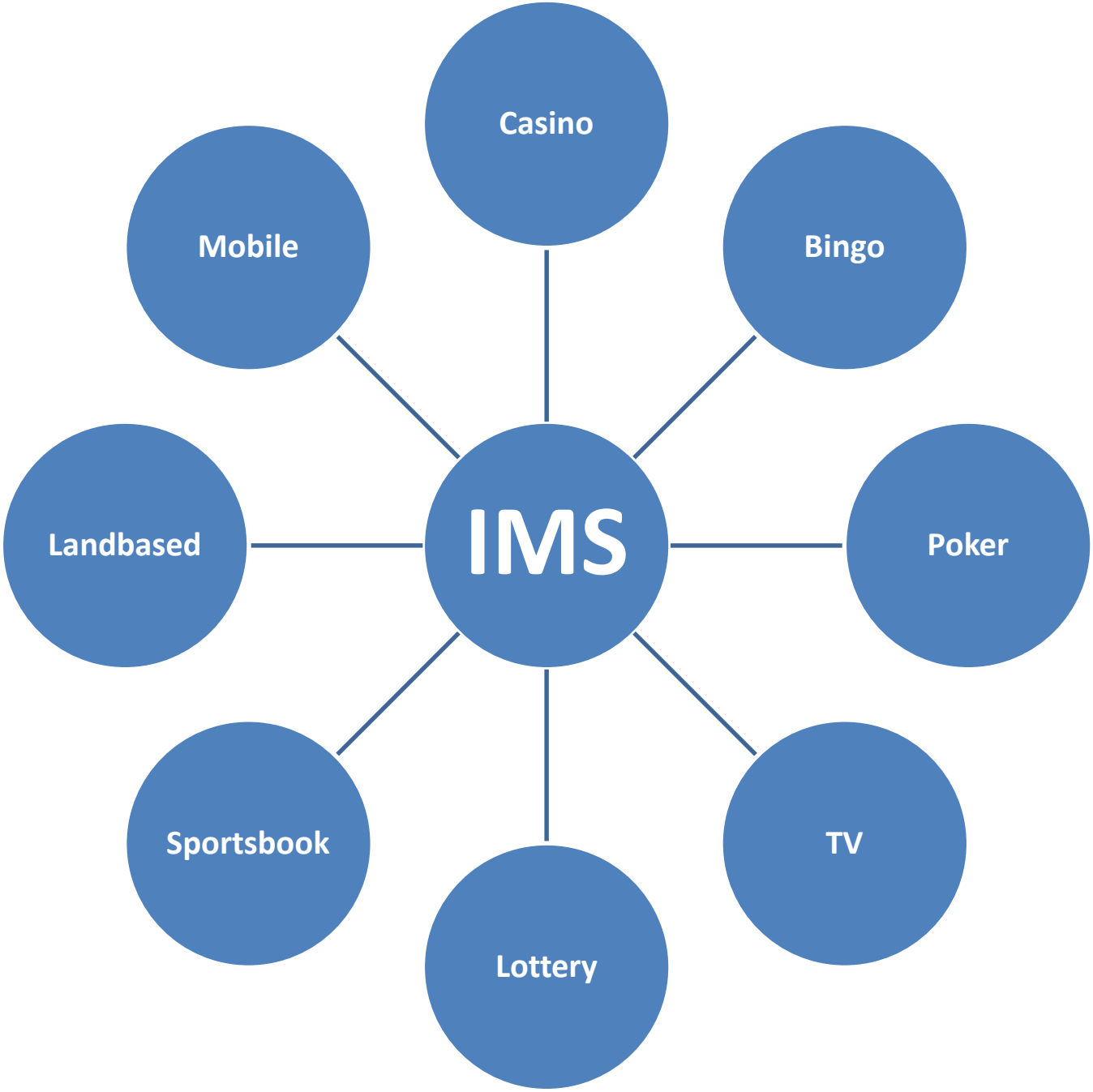
How we test 100 projects each month?

Raul Mäesalu

raul.maesalu@playtech.com

@maedaalu

playtech
SOURCE OF SUCCESS


Casino
Mobile
Bingo
Landbased
IMS
Poker
Sportsbook
TV
Lottery

Numbers

- ~2 million lines of code
- Over 1000 database tables
- ~700 user interfaces for admins
- 386 public API methods
- 95 projects per release
- 290 new defects per release
- ~120 licensees
- ~80 people (including 33 developers, 29 testers)

What is a project

- Any dev request in Jira with an underlying business requirement and:
  - adds new functionalities to our existing system;
  - needs development and testing effort;
  - might have internal or external business needs;
  - might range from hours to weeks of work;
  - one business project might contain several dev requests.

Main risks

- How to...
  - test all new developments on time?
  - deal with regression in a large system like this?
  - support several version branches simultaneously?
  - accommodate the different needs of different customers?
  - handle delays in critical projects?

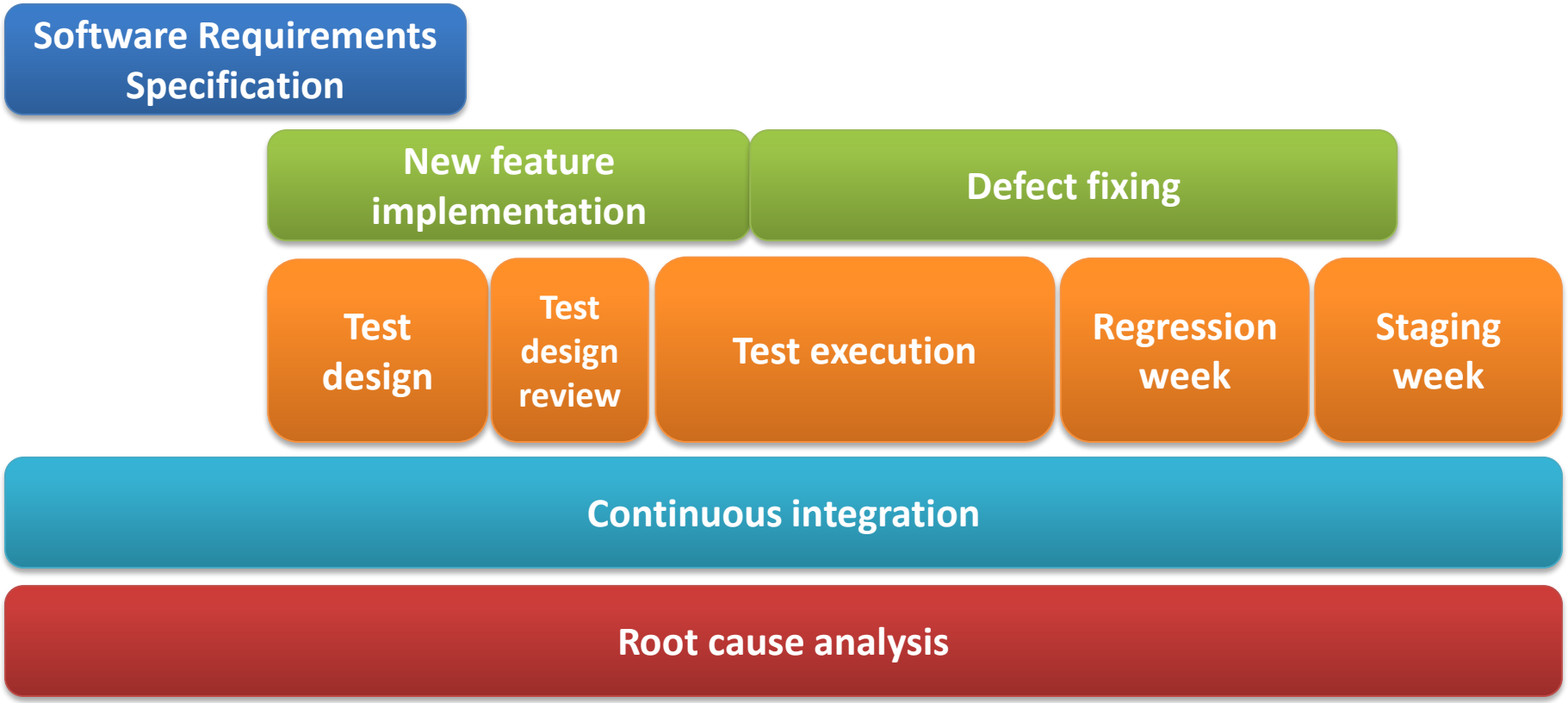
Monthly release cycle

- Monthly major releases supported by weekly maintenance releases
- We have to support 3-6 versions of our software at any given time

Major release structure

- Functional testing period of 2 weeks
  - 65-120 new projects
- Regression period of 1 week
- Staging period of 1 week
- Maintenance release sanity tests each week

The supporting processes



Test design phase

- We write documents called software test plans
- They...
  - contain scenarios instead cases;
  - are strict enough to cover requirements, loose enough to allow creativity and exploring;
  - assume good knowledge of the system;
  - transform into a test report;
  - are input for test automation developers;
  - are reviewed by the analyst and another tester.

Example

Different bonus types and their triggers

Manual bonuses

Bonus type	Completion type	Things to test...	... in combination with following parameters	Issues	Description

Playable (pre-wager)	**Redeemed after wagering**	• Bonus can be assigned
		• Bonus can be wagered
		• On redemption bonus funds + pending winnings are redeemed
		• On removal bonus funds + pending winnings are removed
			• Accept/decline
			• Decline bonus on withdrawal
			• Do not decline on fund transfer withdrawal (if wallet suppors fund transfers to different balances)
			• Different wagering methods:
			  • Bonus
			  • Deposit
			  • Fixed amount
			• Keep winnings pending (if KeepWinningsForced != 1)
			• Required ring-fencing % (if wallet supports ring-fencing)
			• Use bonus money first
			• Display bonus funds separately (with bonus funds and free spins)
			• Show playable games popup after bonus issuing
			• Deposit needed for threshold redemption
			
			All of the above parameters should be tested at least once		

	Never redeemed	• Bonus can be assigned • Bonus can be wagered • When wagering requirements are fulfilled, only pending winnings are redeemed • After the wagering requirements are fulfilled, the bonus acts as a never redeemed bonus without wagering • requirements			
	Never redeemed (without wagering requirements)	• Bonus can be assigned • All winnings that exceed the bonus initial amount are immediately redeemed			
	Immediately redeemed/cash bonus	• Bonus can be assigned • Bonus is immediately redeemed and funds transfered to player's main balance			
After-wager		• Bonus can be assigned • When wagering requirements are fulfilled, bonus funds are redeemed to player's main balance			

Continuous integration

- Code changes are built every night
  - QA owns the build process
- Nightly automated test execution
- It is possible to trigger test execution manually
- Short run test sets
- Separate test automation team

Results example

✗ CI TOTAL TODAY (739) ✗ PASS 735 / FAIL 4 / INCOMPLETE 0		✗ CI MANDATORY TODAY (96) ✗ PASS 95 / FAIL 1 / INCOMPLETE 0	
✓ CI1 playtech420019 ✓ PASS 12 / FAIL 0	✓ CI1 playtech420017 ✓ PASS 2 / FAIL 0	✓ CI1 playtech420016 ✓ PASS 9 / FAIL 0	✓ CI1 playtech420015 ✓ PASS 170 / FAIL 0
✓ CI1 playtech420014 ✓ PASS 11 / FAIL 0	✓ CI1 playtech420013 ✓ PASS 57 / FAIL 0	✓ CI1 playtech420012 ✓ PASS 48 / FAIL 0	✓ CI1 playtech420011 ✓ PASS 92 / FAIL 0
✓ CI1 playtech420010 ✓ PASS 77 / FAIL 0	✓ CI1 playtech42009 ✓ PASS 93 / FAIL 0	✓ CI1 playtech42008 ✓ PASS 37 / FAIL 0	✓ CI1 playtech42007 ✓ PASS 75 / FAIL 0
✗ CI1 playtech42004 ✗ PASS 10 / FAIL 4	✓ CI1 playtech42002 ✓ PASS 9 / FAIL 0	✓ CI1 playtech42001 ✓ PASS 33 / FAIL 0	

Root cause analysis

- Analyse high priority production defects
- Determine root cause
- „How to prevent this situation from happening again?“
- Action items for relevant people and regularly review their progress

The biggest challenges

- Scope changes and late deliveries
- Business critical projects often have very strict deadlines
- Very large projects are a pain to handle
- Legacy stuff
- Test maintenance is time consuming
- Limited time for increasing automation scope
- Version support
- Cross-product projects

Our best solutions

- Don’t panic, embrace the changes
- Be dynamic with your scope, assess your risks and reprioritise accordingly
- Don’t fear failure and make quick decisions
- Reduce bureaucracy by taking ownership of the build process
- Minimise the amount of manual regression
- Ask yourself: what isn’t working well and how to make it better?

Improving the process

- We organise events called department day!
- We choose a relevant theme:
  - previously we have tackled *quality, productivity, gamification, performance*
- We go off site
- Brainstorming and team building
- Identifying the weak spots
- Finding solutions

Notes from our quality day (2010)

- „Introduce automatic nightly builds and continuous integration“
- „Test plan review with analyst & developer“
- „Defect root cause analysis (develop process to make it impossible in future)“
- „More granular and smaller development tasks“

Conclusion

- If you decide to remember something from this talk then I hope it contains the following:
  - Reduce bureaucracy and make quick decisions
  - Document smarter and automate your regression
  - Find ways to improve your process

Q&A

raul.maesalu@playtech.com

@maedaalu