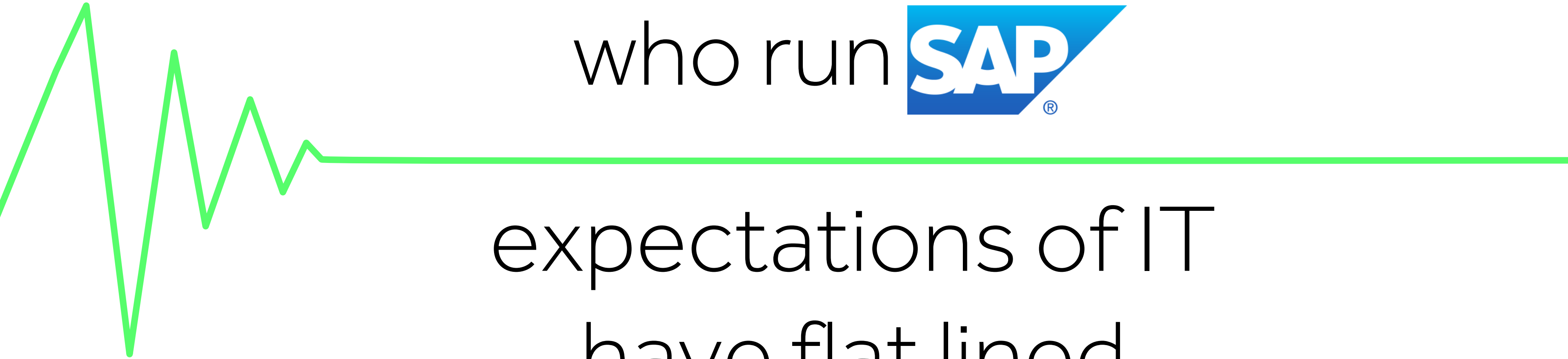
How to Run Agile Development for SAP®

basis

Are you giving
your business
what it really wants?

Or do you have a
:(**business**

In many business
who run SAP®



expectations of IT
have flat lined
and there's a
general feeling of
:(

Agile development means that you deliver the technology that your business needs more quickly

So that your
business becomes
▶ more competitive ◀
fuelling growth and
market share...

And if your business runs SAP then a large proportion of your business processes must run on SAP

But to make your
business more competitive
you need to get
faster at delivering
SAP® projects.

STOP

The secret to becoming more agile is to stop delivering SAP projects.

You need to tell stories instead

of running big projects

Forget that projects ever existed and move to

Continuous Delivery.

Where you constantly deliver smaller business requirements in more frequent cycles.

To do this you need to change the way that you capture **business requirements.**

Forget functional specs and write **user stories.**

you need to tell stories instead

of running big projects

as a [role]
I want to [do something]
so that [value statement]

as a line manager
I want to approve expense claims
so that discretional expenditure
can be controlled

you need to tell stories instead
of running big projects

you need to
tell stories
instead
of running big projects

Don't let your IT team write user stories, have your **business stakeholders** write them instead.

you need to
tell stories
instead
of running big projects

you need to
tell stories
instead
of running big projects

you need to tell stories instead
of running big projects

you need to tell stories instead
of running big projects

Because they'll write them in simple, clear business language.

you need to tell stories instead
of running big projects

you need to tell stories instead
of running big projects

you need to
tell stories
instead
of running big projects

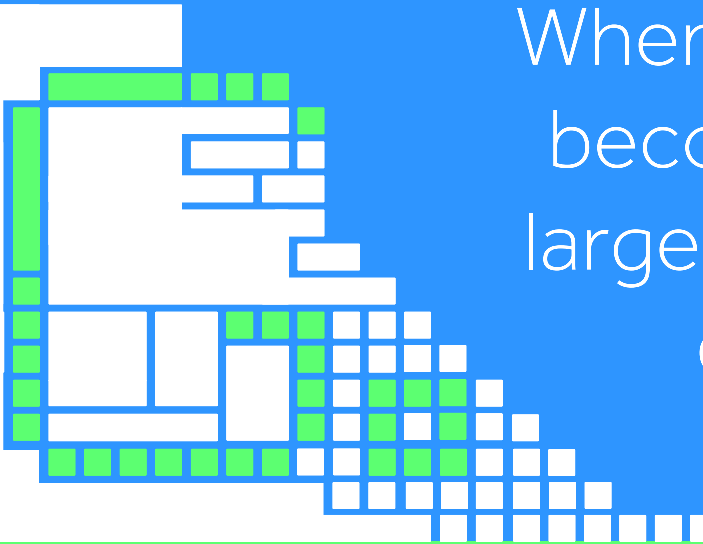
you need to
tell stories
instead
of running big projects

The trick is
to get them to focus
on the what rather
than the how.

you need to
tell stories
instead
of running big projects

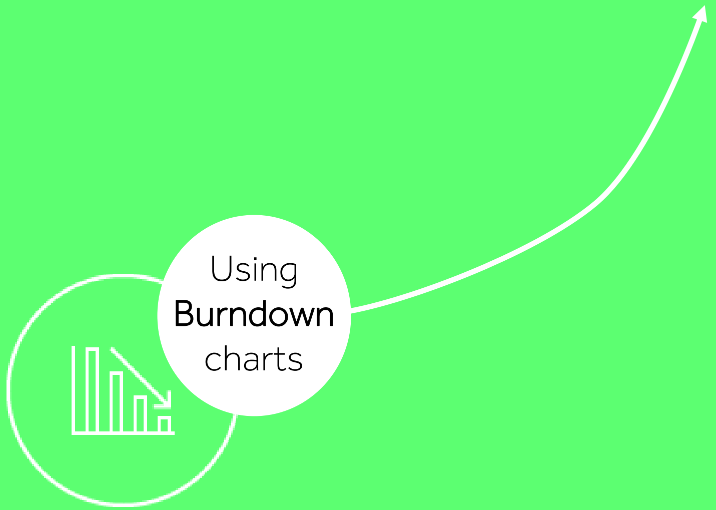
you need to
tell stories
instead
of running big projects

you need to
tell stories
instead
of running big projects

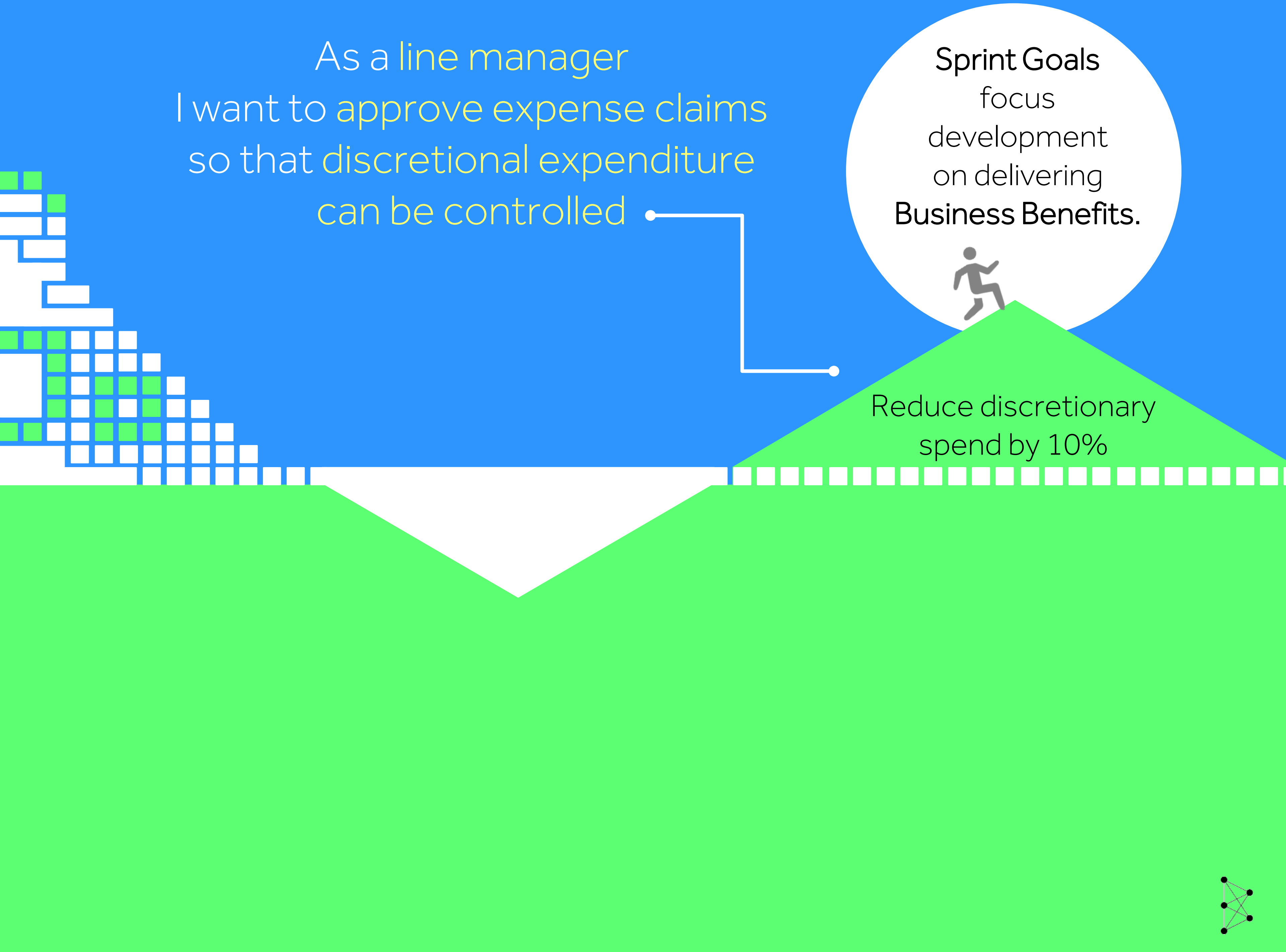


When business requirements become more granular, your larger projects can be broken down and packaged into smaller sprints

user stories



that you deploy on a continuous cycle with defined **shipping dates.**

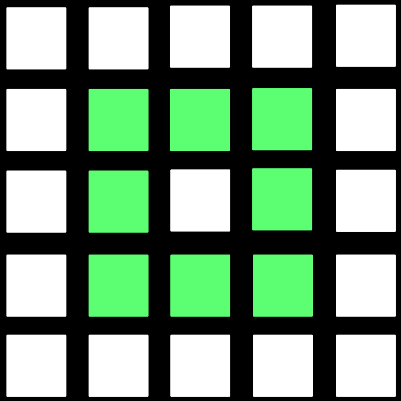


As a line manager
I want to approve expense claims
so that discretional expenditure
can be controlled

Sprint Goals
focus
development
on delivering
Business Benefits.

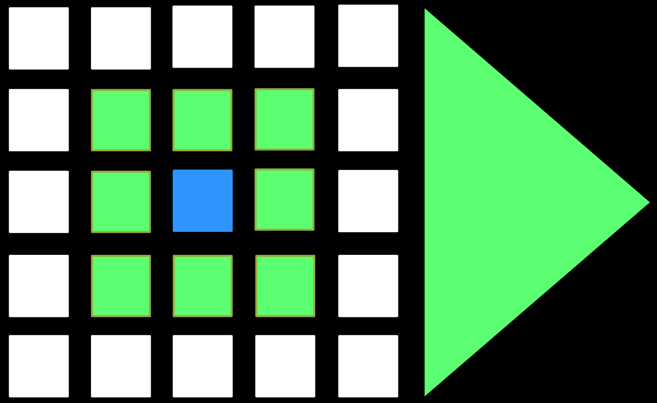
Reduce discretionary
spend by 10%

Sprint bundling



The **bundling** of stories becomes a really important decision - because if integrated stories don’t get finished at the same time, the whole sprint deployment may get delayed.

Continuous Delivery requires Continuous Integration



You need to ensure that the whole sprint can be delivered as a single integrated release.

Or, that incomplete stories can be unbundled late in the sprint without compromising deployment.

Continuous Integration
requires
Continuous Testing

So that you can confidently test the entire deployment **at any time during the sprint.**

You need to create Test environments **quickly, accurately** and be confident that you're testing on the **actual codebase** that's your **candidate for deployment.**

In SAP® terms, this means that transports don't **overtake** or **overwrite** each other.

And, the sequence that transports are deployed to Test is identical to the sequence they'll hit Production.

Plus, you need to be confident that you're testing on data that's as similar to your SAP® Production data as possible.

So you'll need to find ways to quickly copy real SAP® Production data and maybe even scramble sensitive information.

BUT

Your current development process probably involves lots of painful repetitive manual tasks.

And you probably rely on experts* just to make sure things run smoothly.

*expensive people who really ought to be doing more important stuff

“*The key to enabling agile development is the* ***increased use of automation***”

Automating removes manual intervention and **increases development velocity** by freeing developers to focus on **more productive work.**

Automate the approval of development changes

Do your developers wait for days for an e-mail or word document to be sent?

Automate deployment of development changes.

Do your developers sit waiting for some guy somewhere to press a button?

Automate impact assessment of development changes.

Can you honestly guarantee you're testing the right things?

Automate dependency checking of changes within a sprint.

Can you safely perform late unbundling of incomplete code if you need to?

Automate the creation of test environments.

Have you automated your system copy, cloning or virtualization?

Automate regression testing

Is your testing repeatable and most importantly, not dependent on people?

Base test scripts on acceptance criteria that you identify when you create user stories during requirements capture.

you need to tell stories instead

of running big projects

3 days

When you know that you can perform a **full regression test in 3 days at any time during your sprint** you can be confident that your development process is agile.

Traditionally, people treat SAP
as a business system
with business users but
agile SAP development
requires a different mindset.

You have to treat your
SAP solution as a product
that is used by external
paying customers.

Viewing SAP releases as
finished products
that must be shipped
to **paying customers**
creates a mindset shift…

When a release
is shipped it has
to be ready for
public consumption.

It has to be immaculate.

LISTEN

Your SAP® development teams don't talk enough.

Agile development teams need to be more **collaborative.**

Agile thrives on
visibility
and
transparency.

When everybody's on the same
page you can really
increase development velocity.

To iron out problems quickly
and course-correct
during the sprint,
you need
frequent communication.

daily scrum meetings

And to be confident in shipping dates you need accurate development **status reporting.**

burndown charts

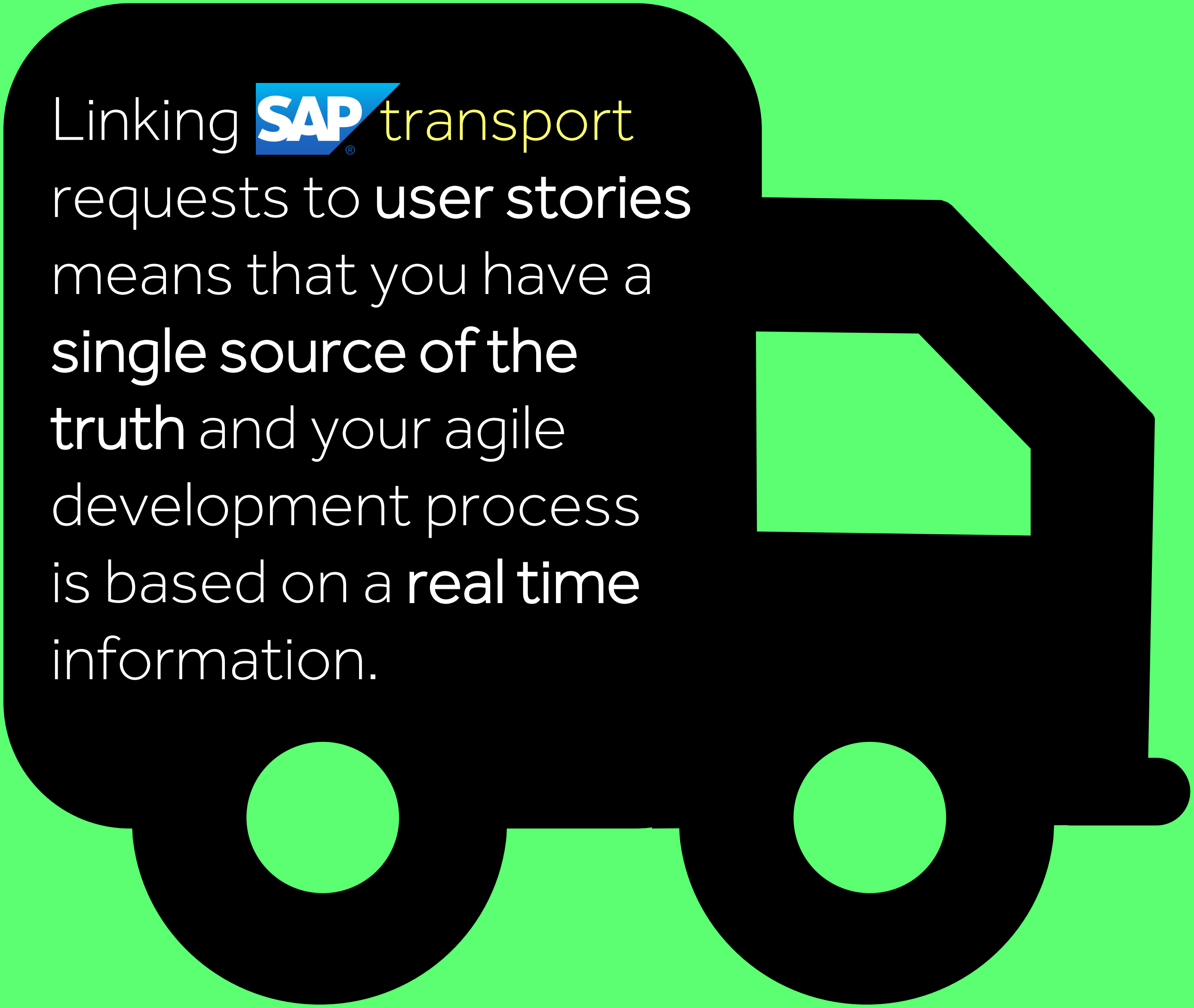
Collaboration doesn't just happen in meetings

Continuous Development

needs

Continuous Collaboration

Each SAP development sits in a **transport** request providing the **perfect vehicle** for you to manage development.



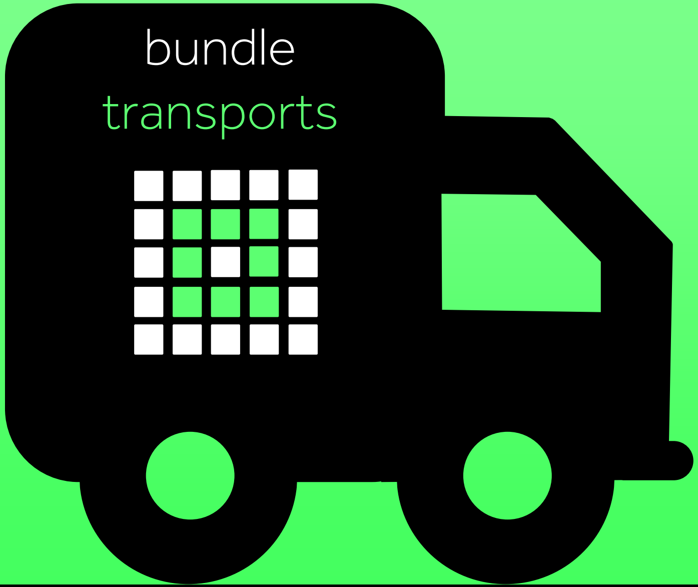
Linking SAP transport requests to **user stories** means that you have a **single source of the truth** and your agile development process is based on a **real time** information.

Link stories to transports

you need to tell stories instead

When each transport relates to a user story you can track delivery of discrete features through your landscape and use the transport to provide real time status of development.

Use advanced change control tools for SAP to record important meta data and group related technical transports together so that you can track product backlog even before the initial transport request is created.



ChaRM warning :
Once a **transport** is
placed into a project
it can't be unbundled.

ChaRM forces bundling
decisions to be made at the
start of your project.

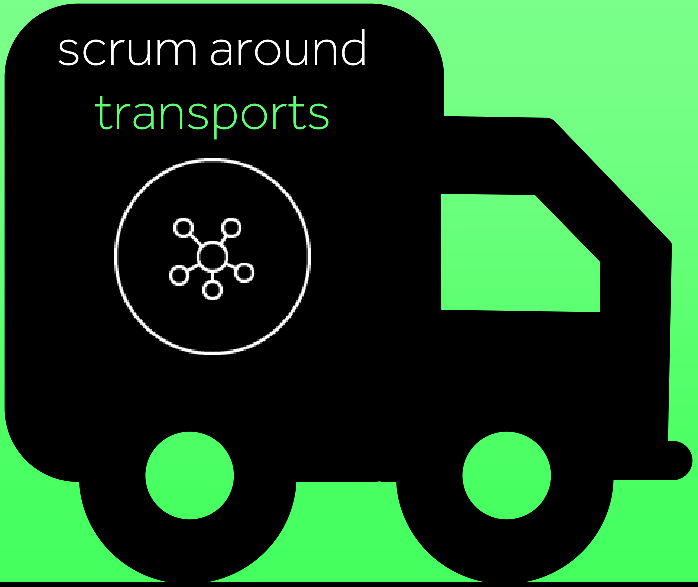
Group transports together so that you can deliver chunks of functionality that logically or technically relate to each other.

Advanced dependency management techniques mean that you can automatically check the objects in your transports and identify whether unbundling is possible.

burndown
transports

If you use transports to track user stories, you can also use transports to manage your delivery burndown.

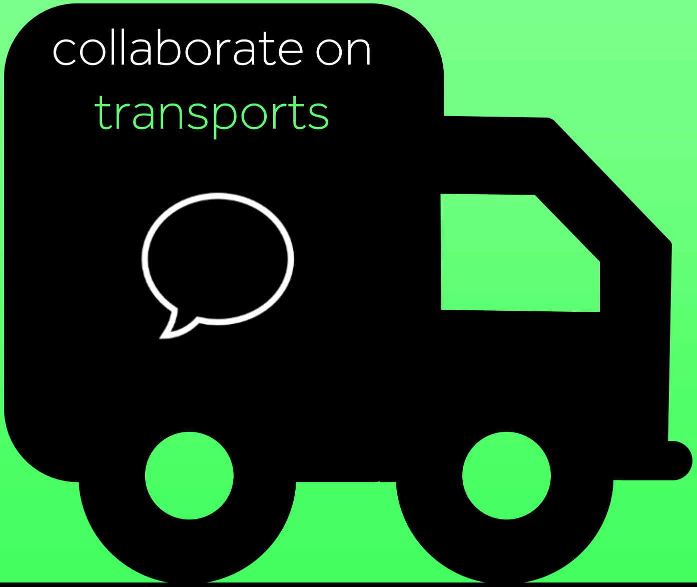
Allocate transports to a sprint and use the status of the transports in your SAP system to monitor progress.



scrum around transports

Your scrum meetings can refer to specific transports during the daily review and because transports are tracked across your SAP landscape in real time there's no need for developers to update an agile software tool.

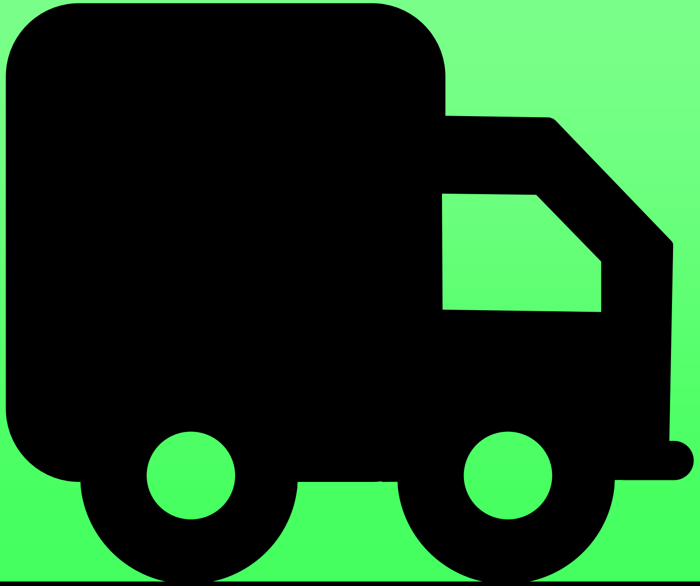
But you may want to look at more advanced add-on transport management tools for SAP



collaborate on transports

Your team gather around transports to collaborate – business and technical, local and remote – the whole team can discuss design, development, delivery and testing of each transport.

Additional meta-data around your transports joins the dots so that your SAP team becomes truly Agile.



Build your agile development process around your SAP transports and start shipping serious value.

basis

A faster, better way

basistechnologies.com