

estimating tasks

lisa komidar

IT project engineer manager, fusionstorm

lkomidar@fusionstorm.com

rose pruyne

IT project manager, penn state

rtp2@psu.edu

overview

- i. high-level estimating
- ii. time-based estimates
- iii. effort+complexity-based estimating with story points
- iv. planning poker for fun and profit

high-level estimating

high-level estimating happens **at the beginning** of the project, project stage, or release planning.

the chief purpose is to **prioritize** work.

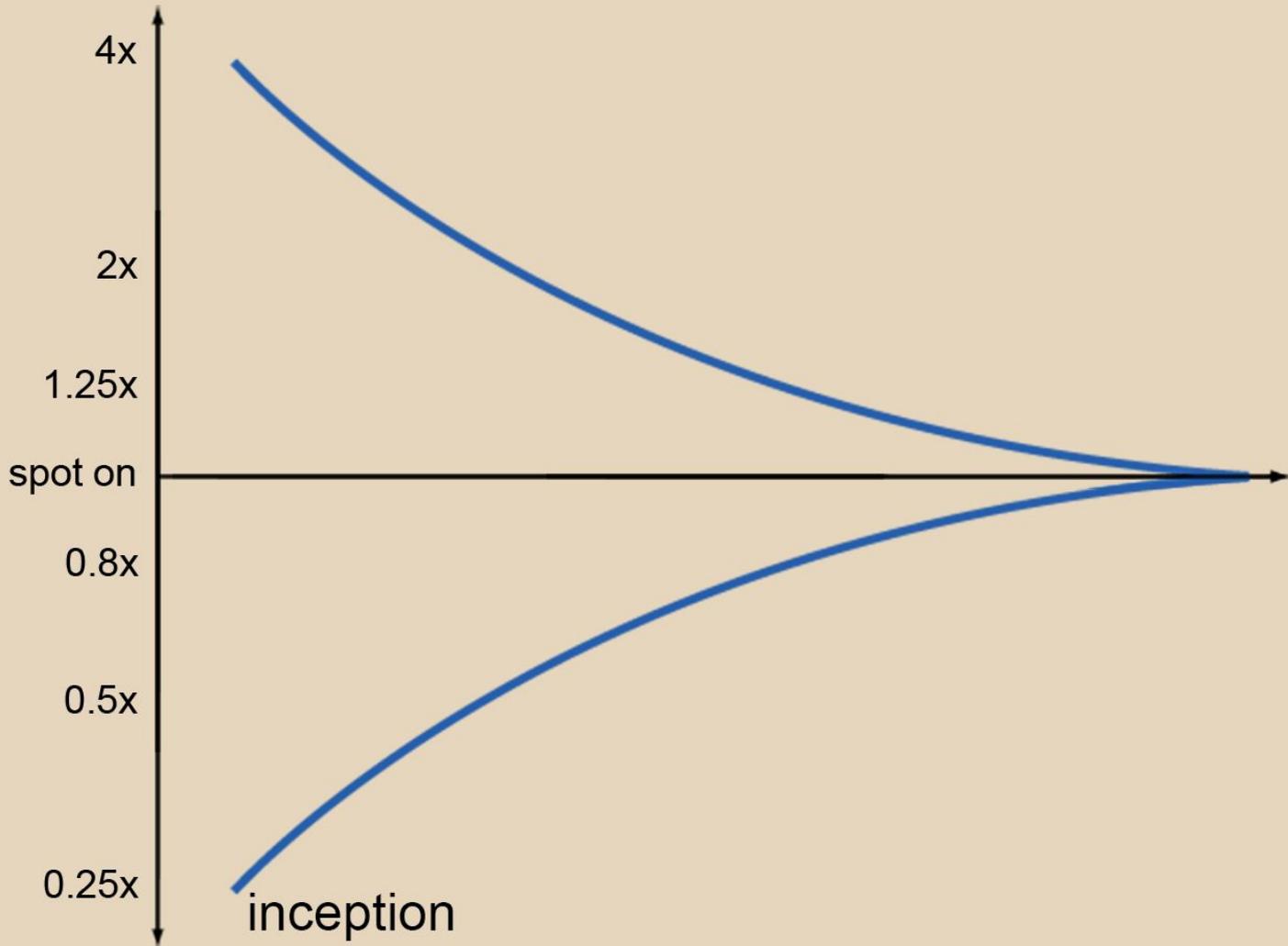
idenfiy **resource** needs.

ballpark.

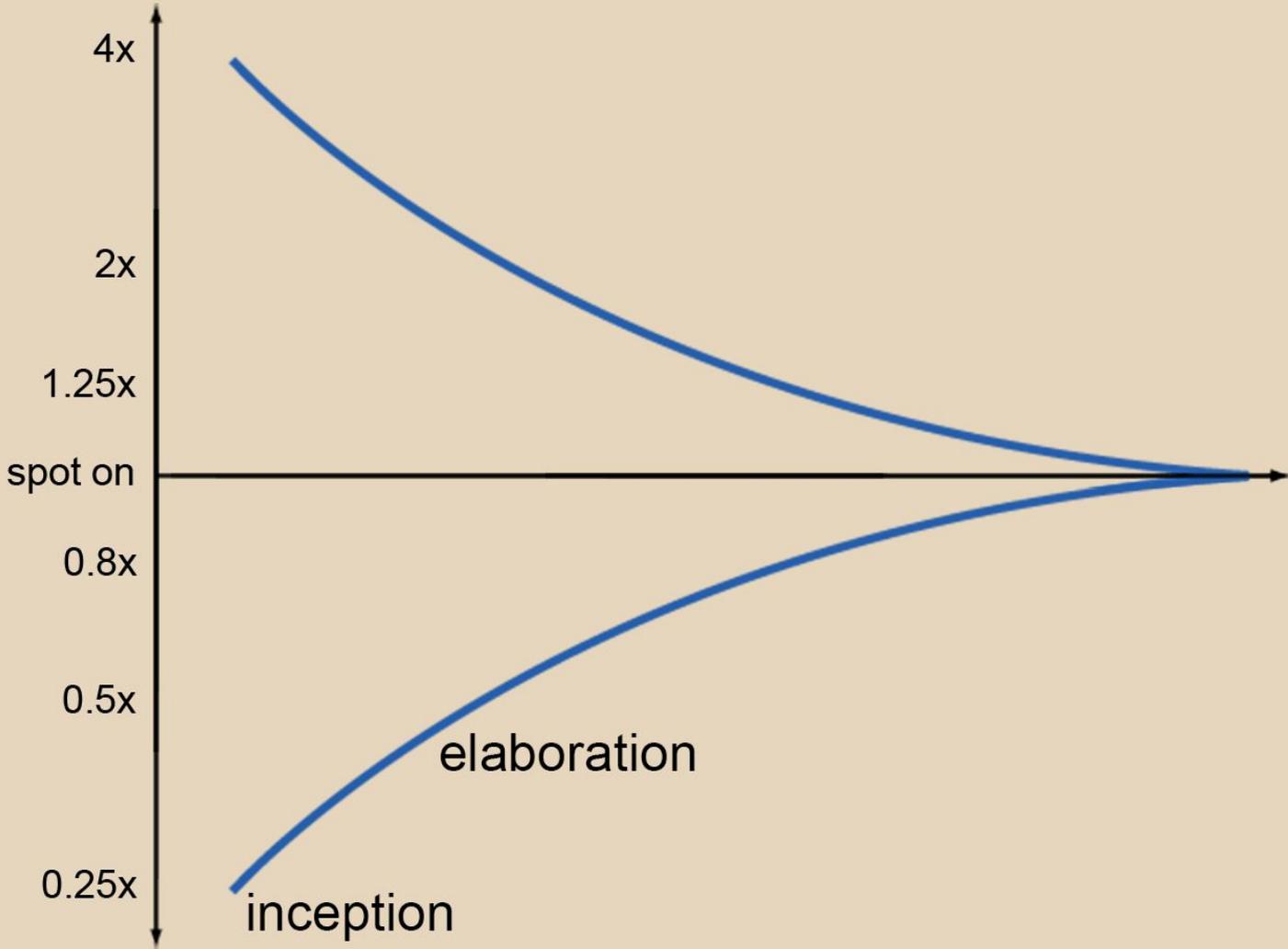
it is **not** about making promises.

a little problem with **waterfall**.

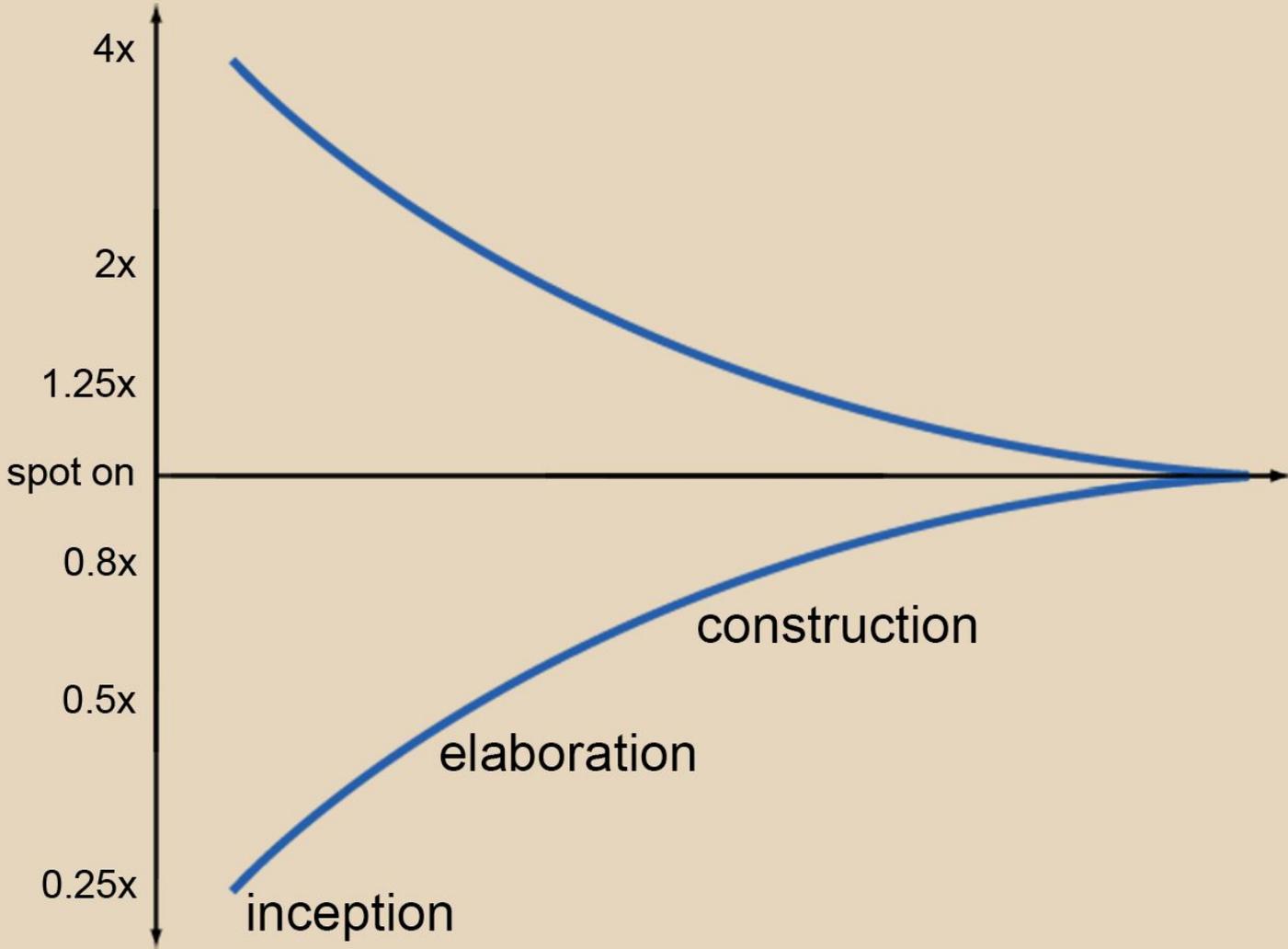
variation of estimation accuracy



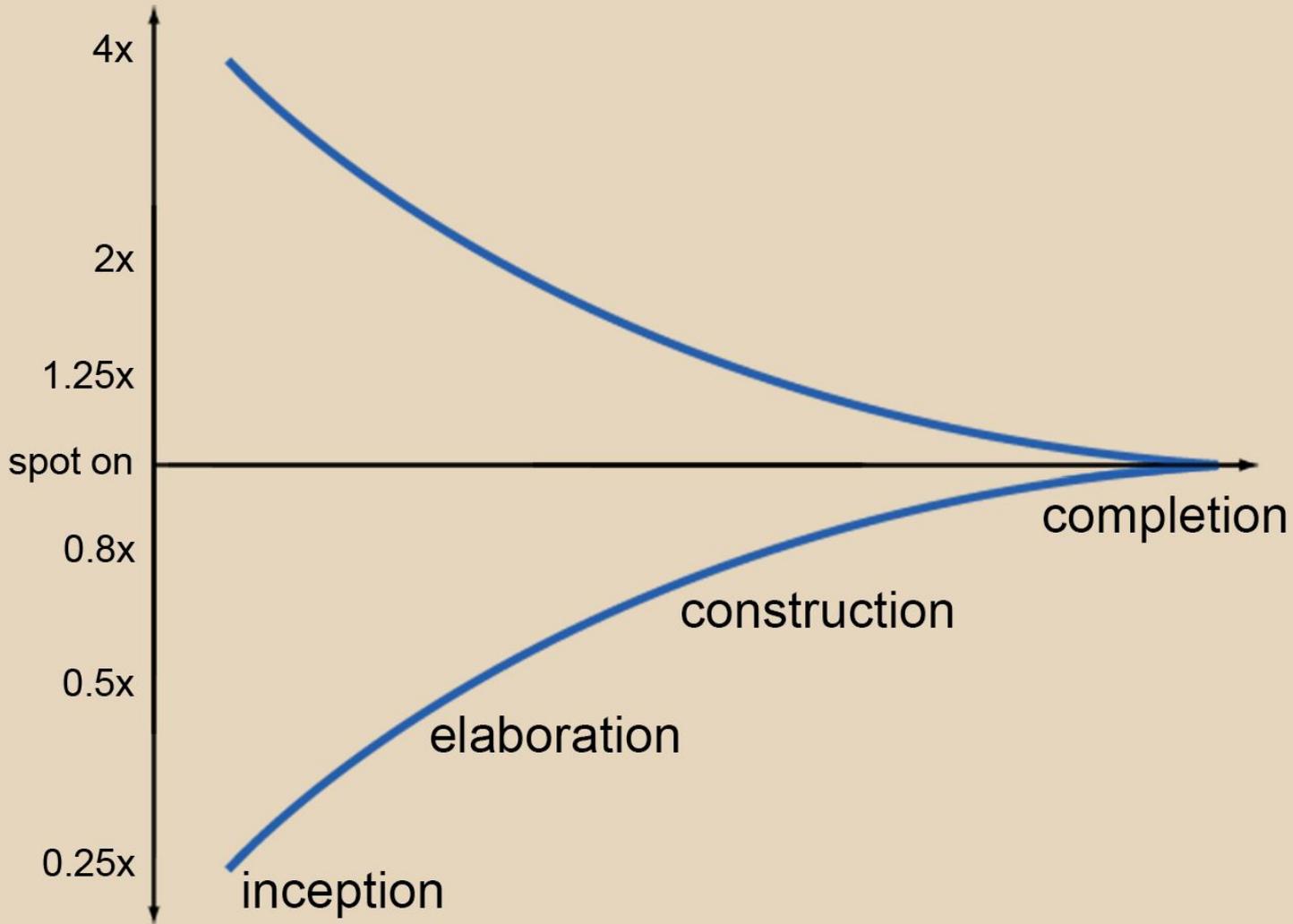
variation of estimation accuracy



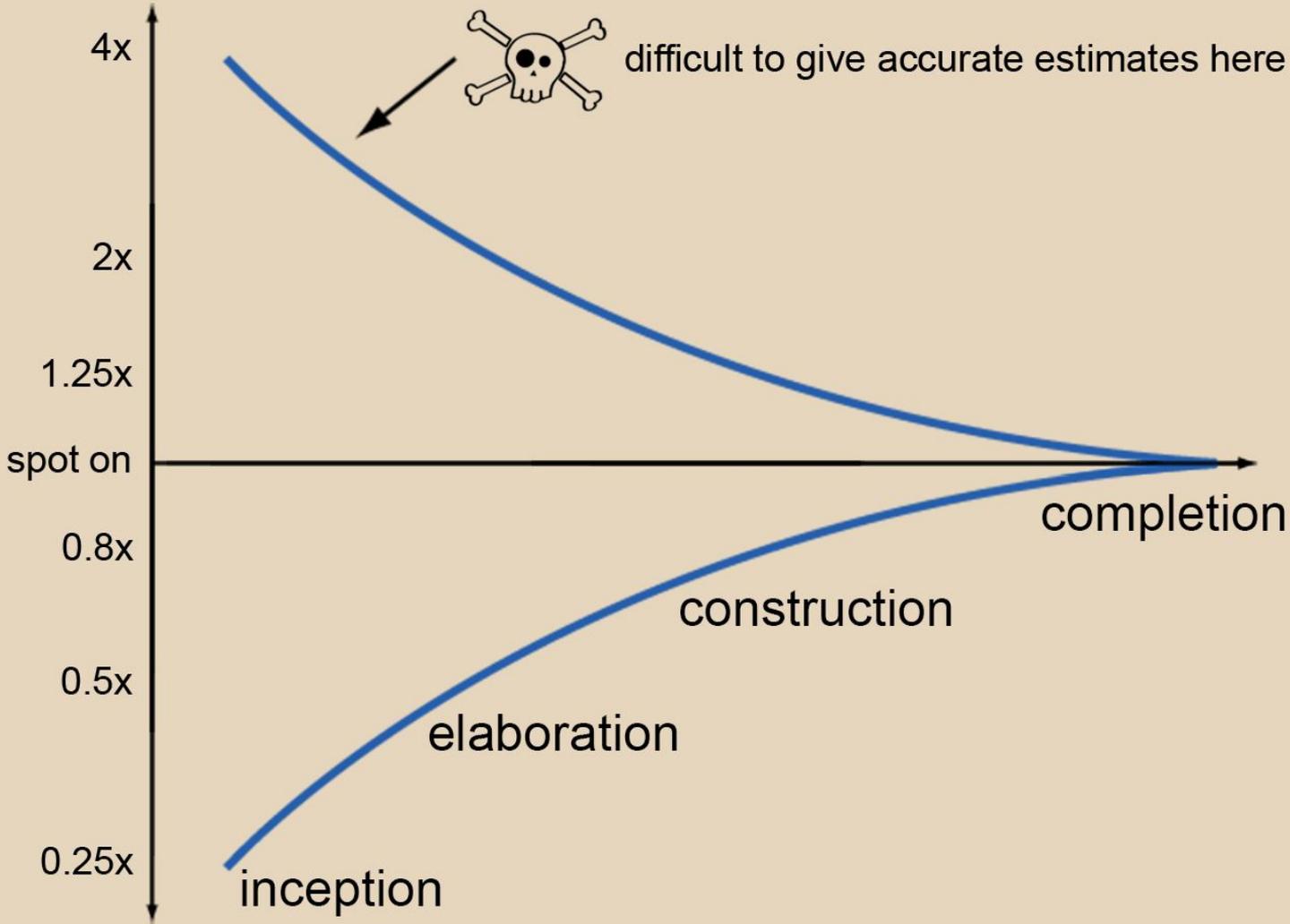
variation of estimation accuracy



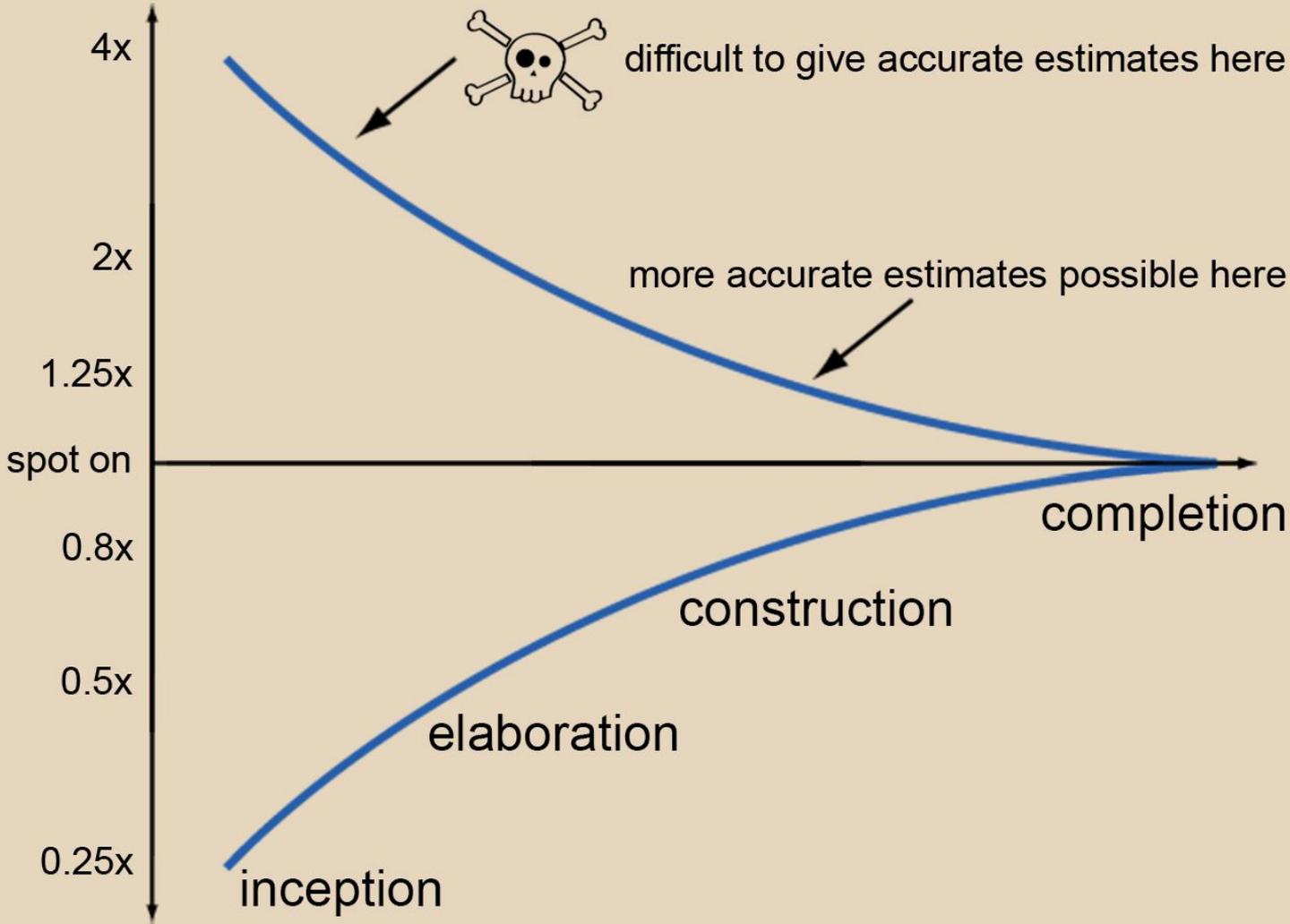
variation of estimation accuracy



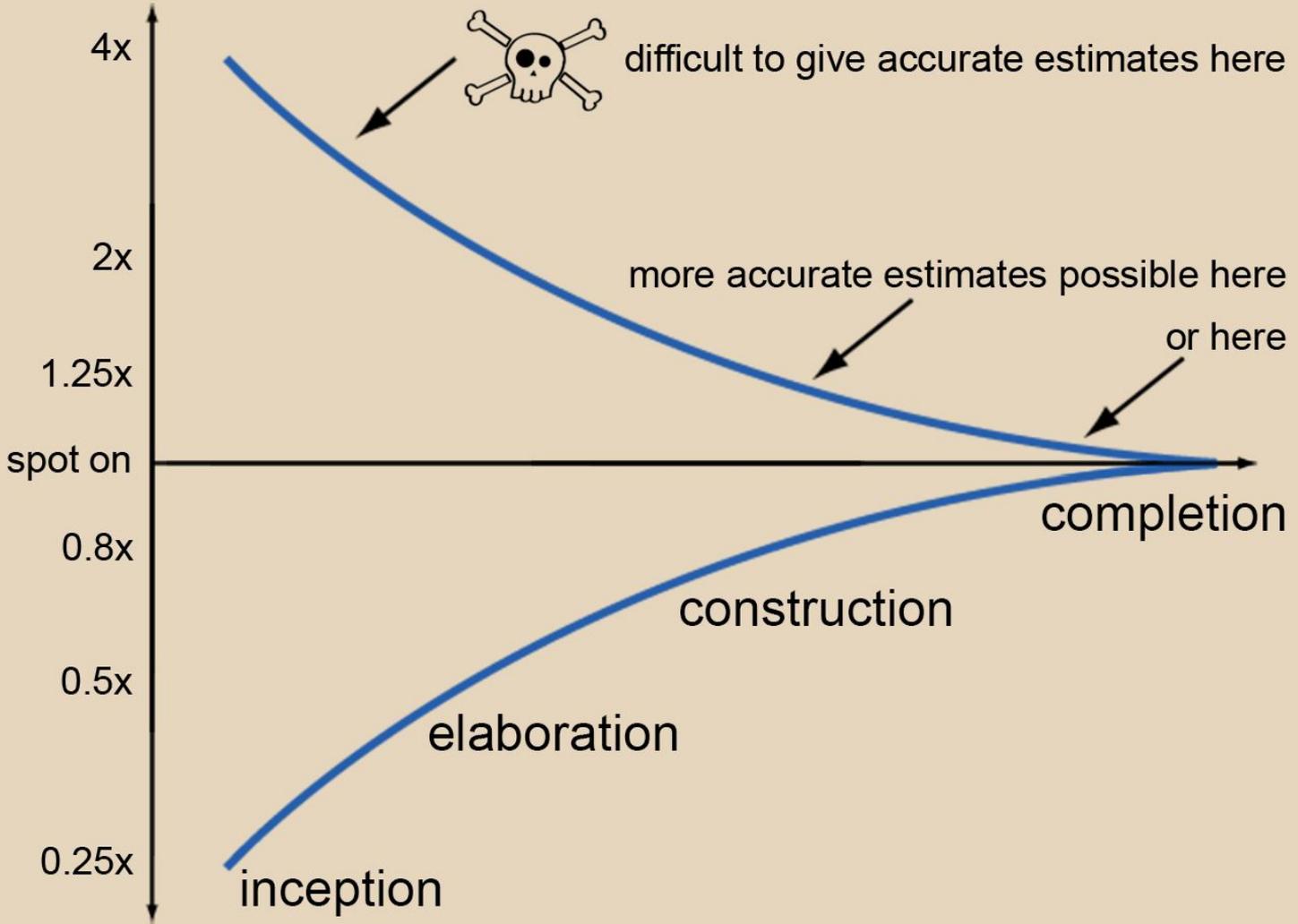
variation of estimation accuracy



variation of estimation accuracy

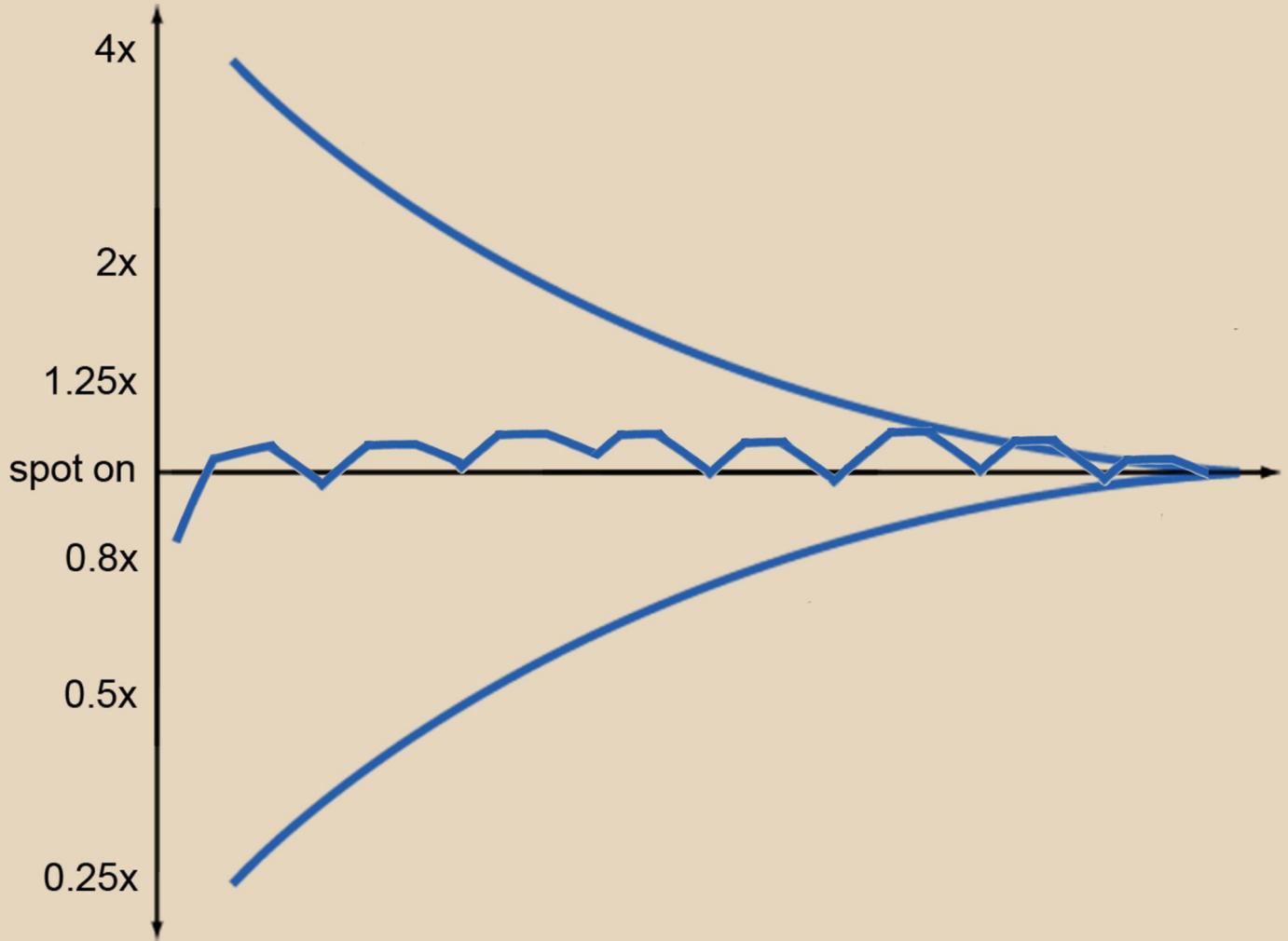


variation of estimation accuracy



an **iterative** approach works better for **change-driven** product development.

variation of estimation accuracy



high-level estimates have significant variance.

make your estimations at the **feature** level...

...then do the **math**.

a feature is a chunk of functionality that delivers **business value**.

"users need to be able to comment on articles."

20 story points

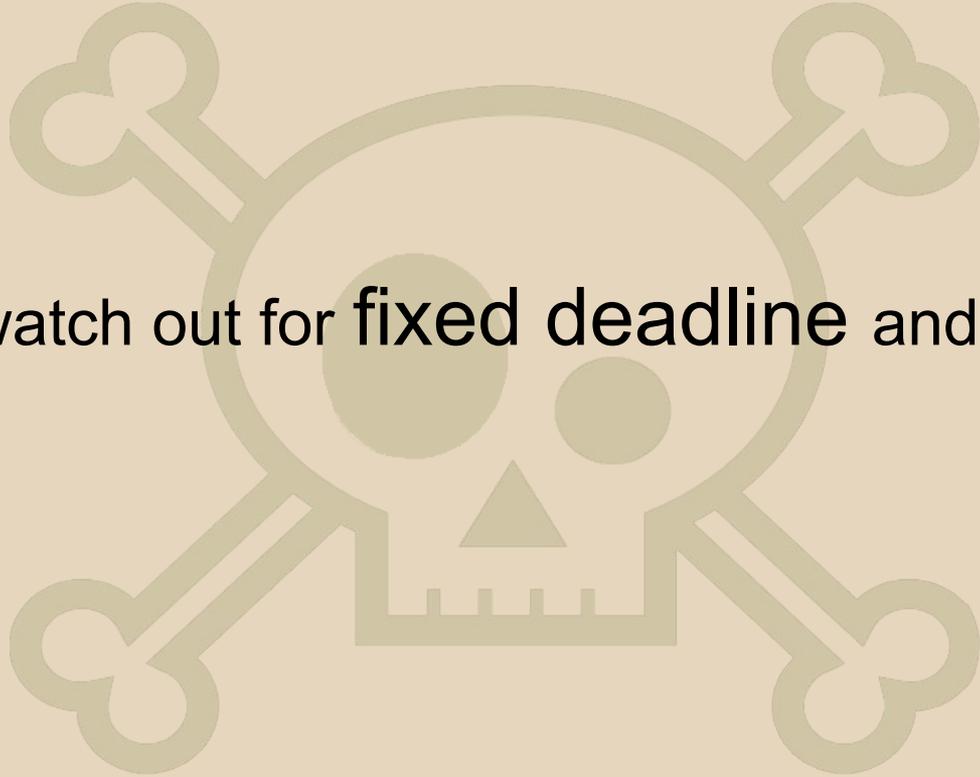


accept that you will need to **adjust the accuracy.**



80 hours

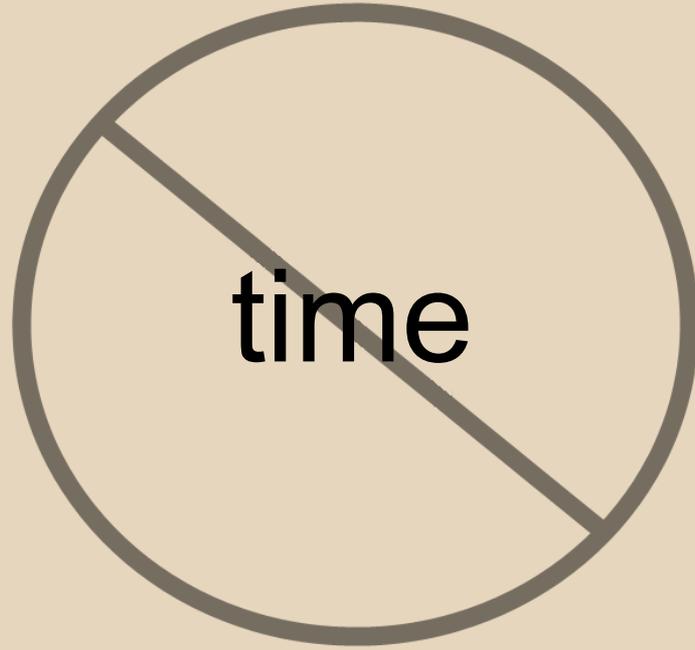
watch out for requirements that are
impractical.
poorly understood by the
customer.
risky.
unclear or confusing.
include hidden work.



especially watch out for **fixed deadline** and **fixed scope**.

time-based estimates

traditional estimates are based on **time**.



time

pm: so, how long do you think it will take to finish this, um, thing?

developer: oh, I don't know. 4 to 6 weeks. maybe 8.

pm: so, how long do you think it will take to finish this, um, thing?

developer: oh, I don't know. 4 to 6 weeks. maybe 8.


optimistic


not so optimistic


pessimistic

what the pm concludes: $(4 + 6 + 8)/3 = 6$ weeks


150%

6 weeks



no one trusts this estimate.

6 weeks



**no one trusts this estimate.
it's probably wrong.**

time estimates usually are overly **optimistic**.



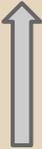
**forced by fixed scope &
deadline (very bad idea)**

reluctance to give bad news

missing unexpected complexities

estimated way too early

or they are overly pessimistic.



padding

time estimates do not take into account the amount of work a team can complete within a given time.

story points

"users need to be able to comment on articles."

smaller aspects of this feature:

as somebody commenting on an article, i need to be able to edit my comment so that i can fix errors and not appear stupid.

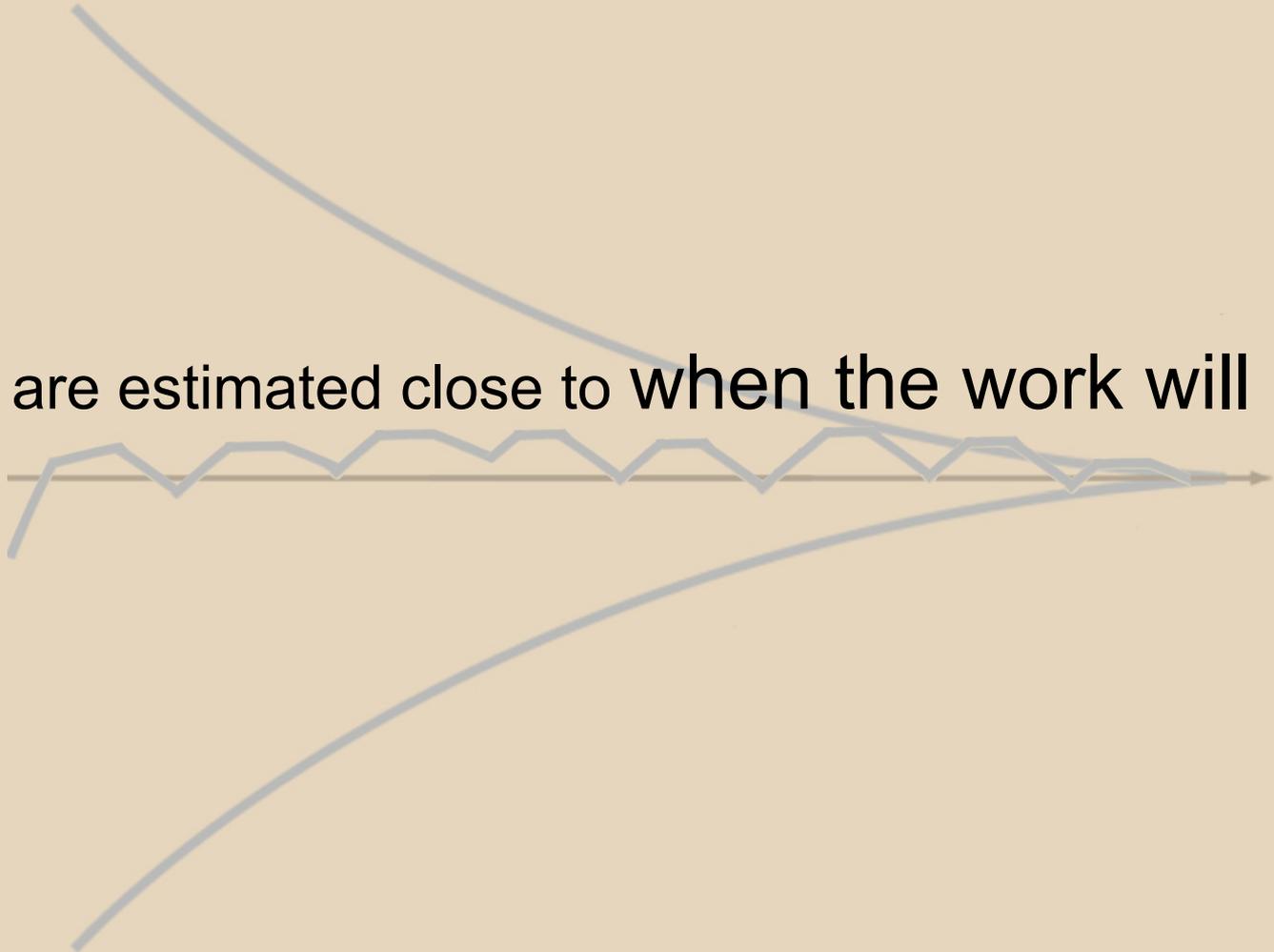
as an author, i'd like readers to be able to rate comments so that i can see which comments on my articles are the most popular.

as the site admin, i need to stop bots from spamming comments.

story points are an alternate way to estimate when work will be done.

story points are based on **effort** and **complexity**.

story points are estimated close to **when the work will be done.**



story points are tailored to the uniqueness of each team.

the team **agrees** on what a story point equals.

1 point might be...fix a minor css error.

participate in a routine project work session.

read a blog article.

2 points might be...write instructions on something you know.
fix 3 or 4 relatively minor coding errors.

3 points might be...write some instructions that need research.

5 points might be...plan and give a 1-hour presentation from scratch.

0, $\frac{1}{2}$, 1, 2, 3, 5, 8, 13, 20, 40, 100

so, how does this **help** with estimation?

consider this scenario...

your team

is consistent in estimating points.

completes, on average, 30 points every 2 weeks.

has estimated all the stories in a feature.

that estimate equals 180 points.

this is called **velocity**.



your team

is consistent in estimating points

completes, on average, **30 points** every **2 weeks**

has estimated **all the stories** in a feature

that estimation equals **180 points**

therefore,

based on a **velocity** of **30**, your team can **complete** all the work in **6** iterations, or **12 weeks**.

story points

provide **more accurate** estimations

provide **stable** estimations based on performance data

allow your team to achieve a **steady state**

story points

help to **manage expectations.**

help to **avoid** scary promises.

give you and your team more **credibility.**

story points

help teams improve performance.

give teams confidence and motivation.

Playing Poker for Estimating Stories

further reading

- [today's trello board](#)
- [an introduction to story points](#)
- [the origin of story points](#)
- [story points: why are they better than hours?](#)
- [9 reasons why you should use story points](#)
- [story points versus hours](#)
- [story points are still about effort](#)
- [common project management metrics doom IT departments to failure](#)

lisa komidar

IT project engineer manager, fusionstorm

lkomidar@fusionstorm.com

rose pruyne

IT project manager, penn state

rtp2@psu.edu

this presentation can be found at
goo.gl/agCVmo