Fedora Minimization Objective — Logic Model

① Resources/Inputs	② Activities	③ Outputs	④ Outcomes	⑤ Impact
What we have: time, money, code, hardware,...	What we do with our resources	The direct products of our activities	The specific change in the world due to those outputs	Mission and vision; our long-term effect

Things we can affect directly | | | **Out of our direct control** |

Resources/Inputs	Activities	Outputs	Outcomes	Impact
infrastructure to prototype and run services experienced maintainers to help with minimization devs interested in developing service prototypes travel budget	build monitoring services identify use cases and allow community to define other ones using ↑, explore what can be minimized and do it engage with upstreams for bigger changes blog and talk at events about what's going on meet in person to hack and set direction implement policy changes for bigger changes, such as an approach for the "containers vs. system" problem	The Fedora community focuses on specific use cases (and the infra supports it) for: • testing • size monitoring and automated bug reporting • minimization, both making and keeping small Use case definitions help wit prioritizing limited resources to focus on what has been recognized (and transparently defined) as important for the project.	open source devs can focus on features because ops use packaged stuff and don't ask for immediate CVE fixes etc. ops can run stuff easily because it's packaged, while being cutting edge, and optimal for their use case in terms of size and features. Getting all the benefits as updates, easy installation etc. fedora maintainers can focus on relevant stuff thanks to automation, monitoring, tools, and clear(er) priorities thanks to use cases	**Vision:** Thousands of individual and corporate contributors collaborate in the Fedora community to explore new problems and to build a fast-moving modern OS with a rich ecosystem allowing them to experiment on modernising their infrastructure. **Mission:** "Helping open source devs, ops, (and maintainers) focus on what is relevant for them by taking pressure from devs and making software convenient to use for ops."