Home Assistant

Ali Sina (218318428)
Arash Saffari (218791632)
David Luu (216157463)
John Donato Prabahar (219087279)
Omer Omer (218636878)
Siyan Sriganeshan (218707190)

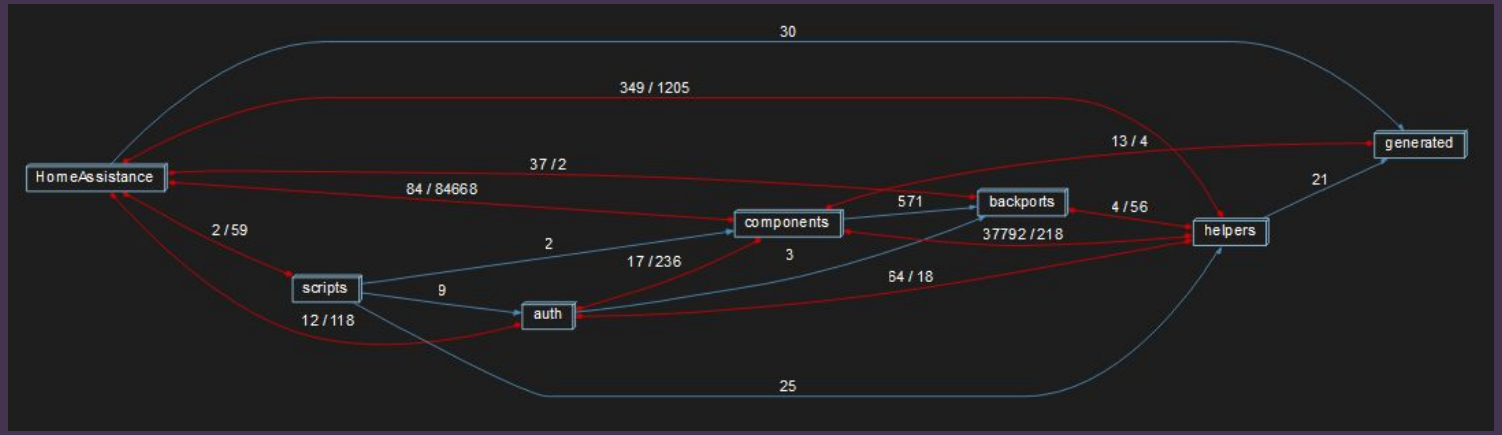
EECS 4314

01

Derivation Process

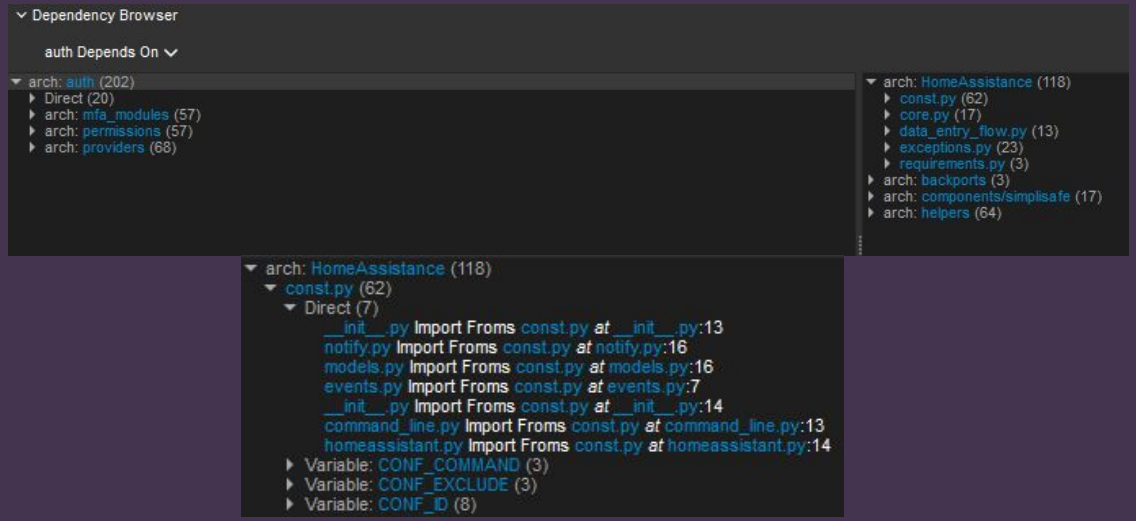
Derivation Process

- Event Bus, Service Registry, and State Machine were all found in core.py, so it was all placed in an entity called Home Assistant



Derivation Cont.

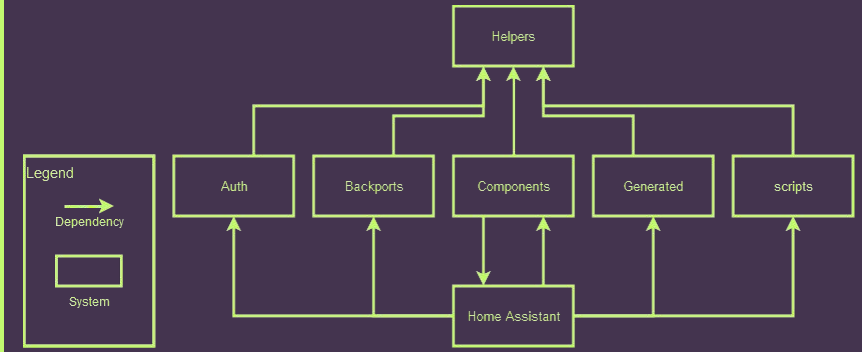
- The dependencies were than examined using the depadancy browser
- Would could then examine the code and determine the four W's

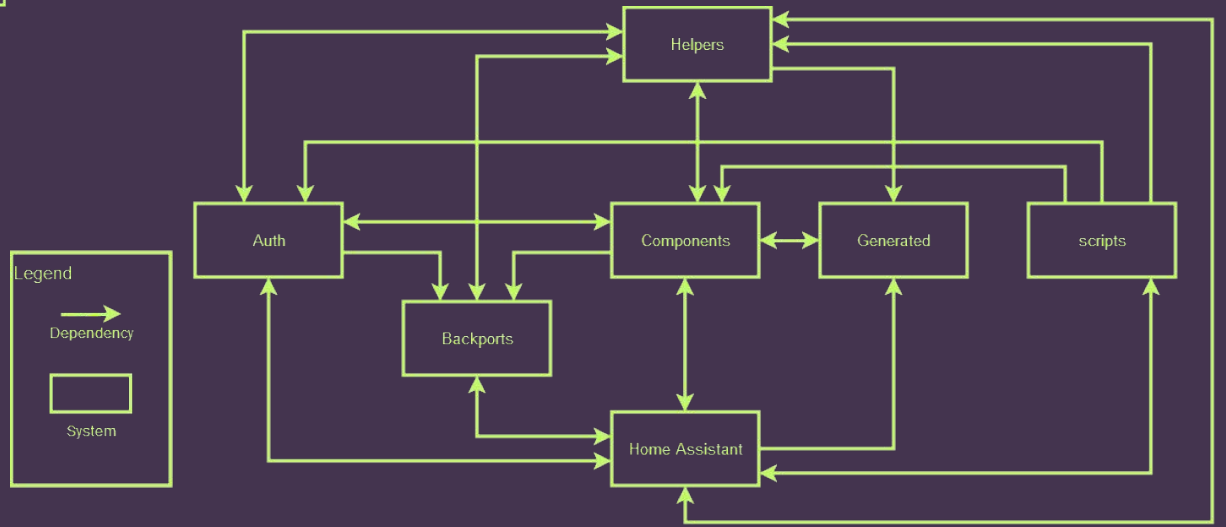


02

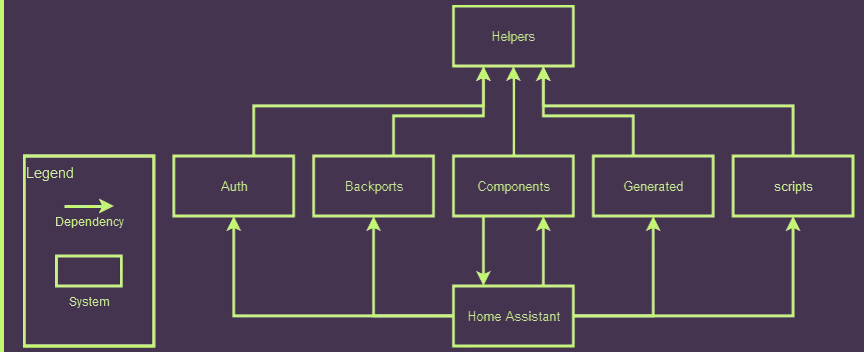
Revised Dependencies

Conceptual To Concrete

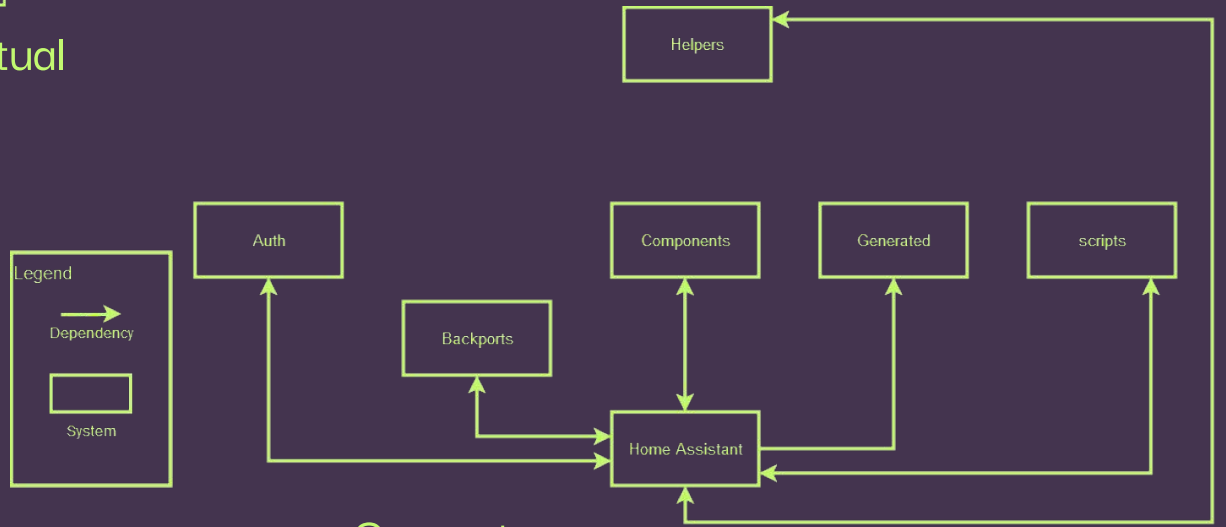




Divergences Of Home Assistant



Conceptual

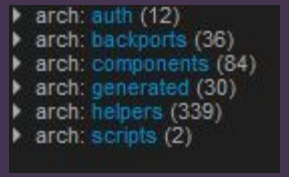


Concrete

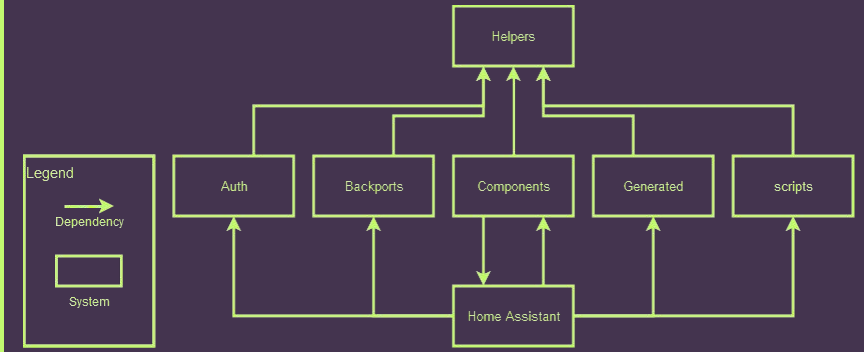
Home Assistant Cont.

- Callback Function
- Determines whether method is safe to call within event loop
- Has most dependencies in helper entity

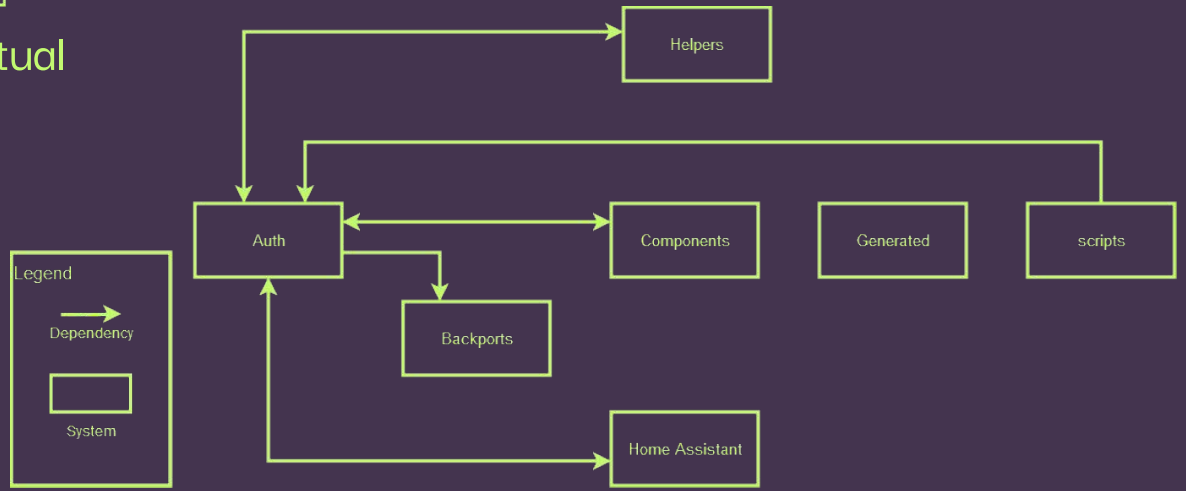
```python
def callback[_CallableT: Callable[..., Any]](func: _CallableT) -> _CallableT:
    """Annotation to mark method as safe to call from within the event loop."""
    setattr(func, "_hass_callback", True)
    return func
```



Divergences Of Auth



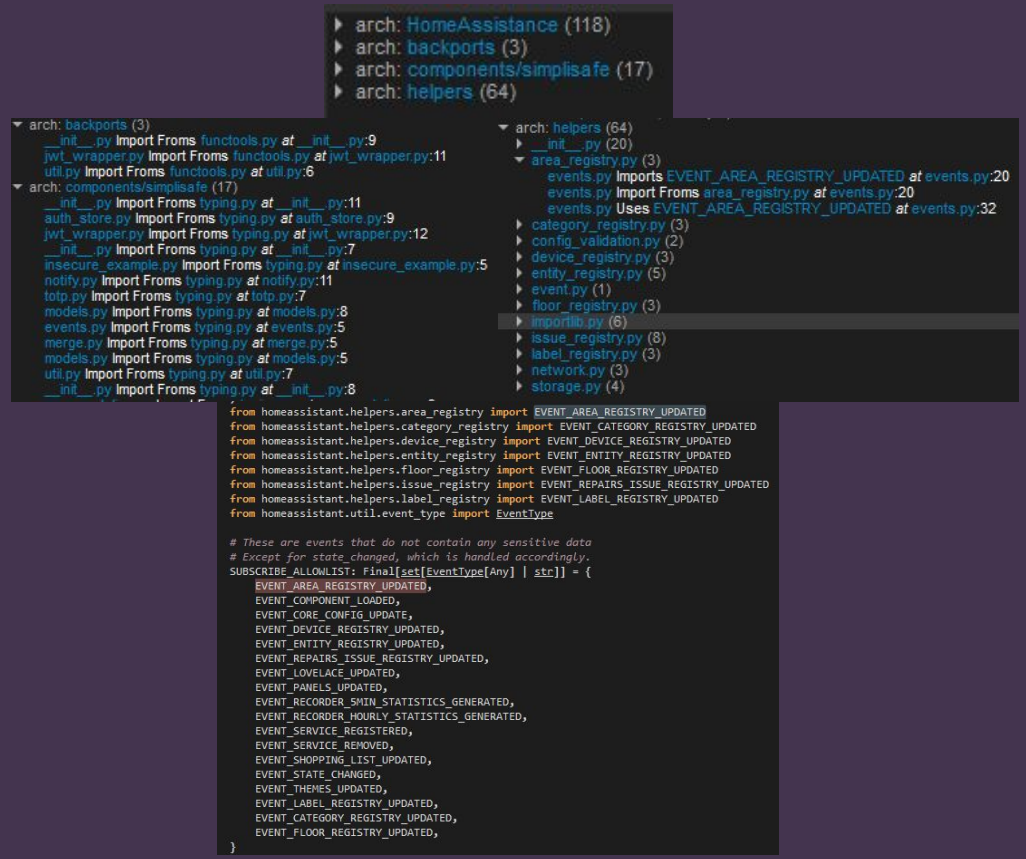
Conceptual



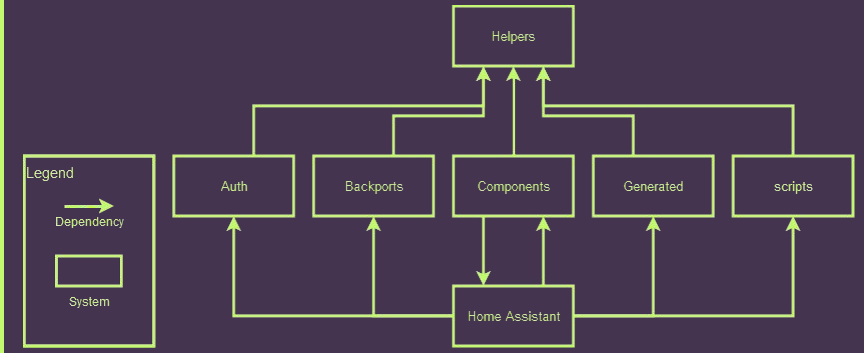
Concrete

Auth Cont.

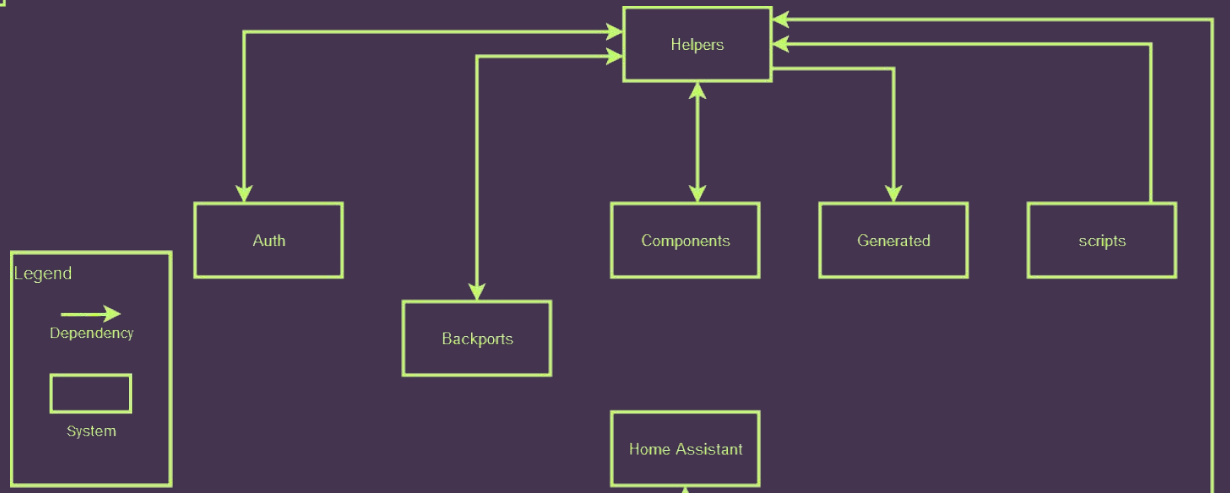
- Mainly imports and uses from variables and classes, mainly from Home Assistant
- Most classes would need auth to make sure proper users access it



Divergences Of Helpers



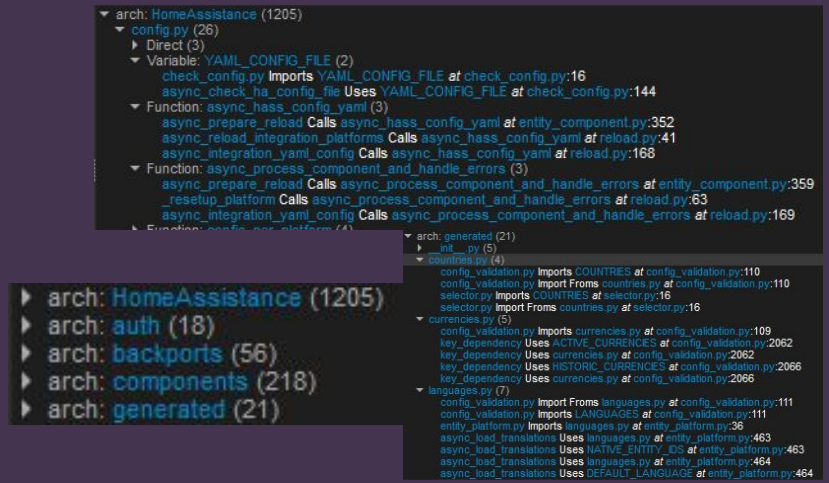
Conceptual



Concrete

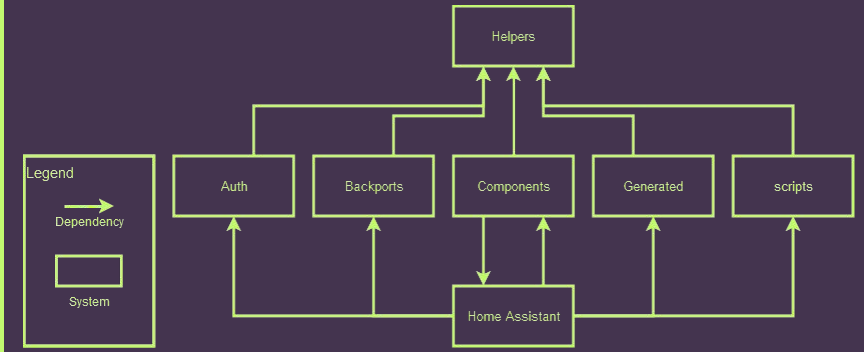
Helpers Cont.

- Helpers imports data like currencies, countries, and languages from generated files for error checking
- By importing helpers can quickly access that data whenever needed

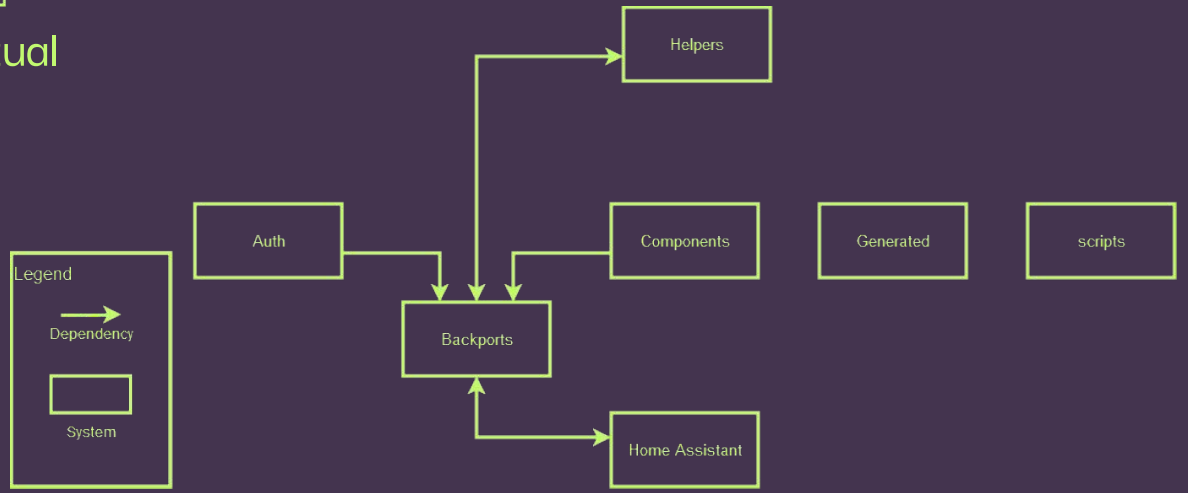



```
from homeassistant.exceptions import HomeAssistantError, TemplateError
from homeassistant.generated import currencies
from homeassistant.generated.countries import COUNTRIES
from homeassistant.generated.languages import LANGUAGES
from homeassistant.util import raise_if_invalid_path, slugify as util_slugify
import homeassistant.util.dt as dt_util
from homeassistant.util.yaml.objects import NodeStrClass
currency = vol.In(
    currencies.ACTIVE_CURRENCIES, msg="invalid ISO 4217 formatted currency"
)

historic_currency = vol.In(
    currencies.HISTORIC_CURRENCIES, msg="invalid ISO 4217 formatted historic currency"
)

country = vol.In(COUNTRIES, msg="invalid ISO 3166 formatted country")

language = vol.In(LANGUAGES, msg="invalid RFC 5646 formatted language")
```

Divergences Of Backports



Conceptual

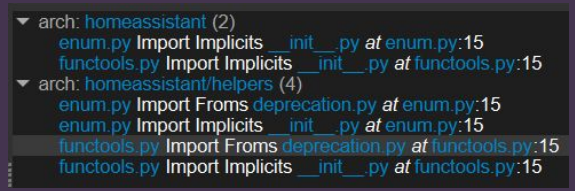


Concrete

Backports Cont.

DeprecatedAlias

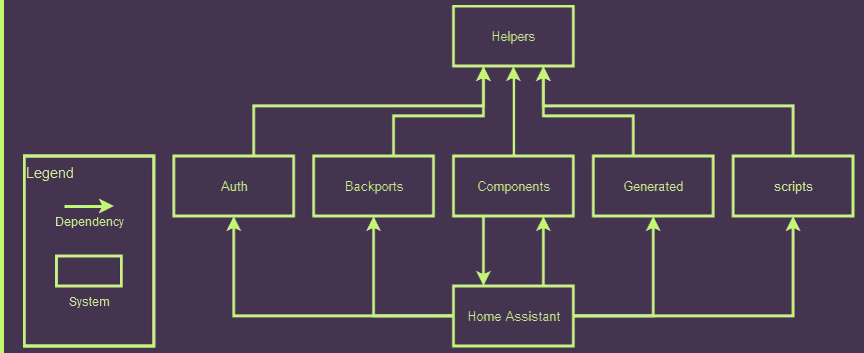
- Tracks Deprecation
- Replacement Suggestion
- Timeline for Depreciation
- Warns Users



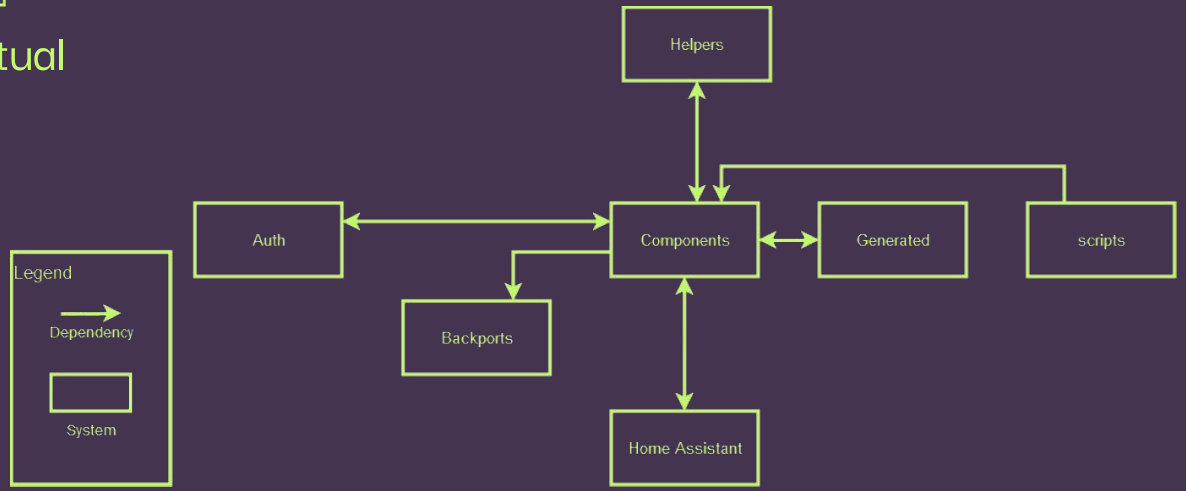

```
from homeassistant.helpers.deprecation import (
    DeprecatedAlias,
    all_with_deprecated_constants,
    check_if_deprecated_constant,
    dir_with_deprecated_constants,
)
```

```
# cached_property deprecated as of 2024.5 use functools.cached_property instead.
_DEPRECATED_cached_property = DeprecatedAlias(
    _cached_property, "functools.cached_property", "2025.5"
)
```

Divergences Of Components



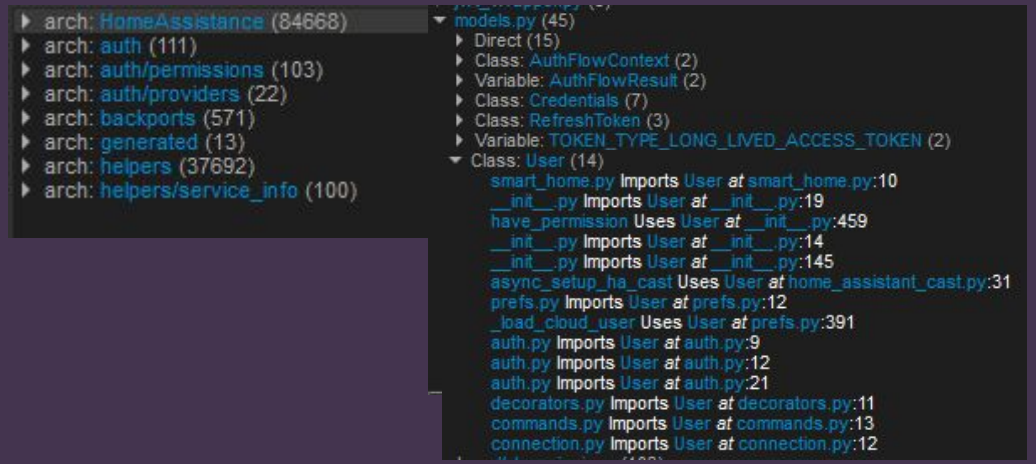
Conceptual



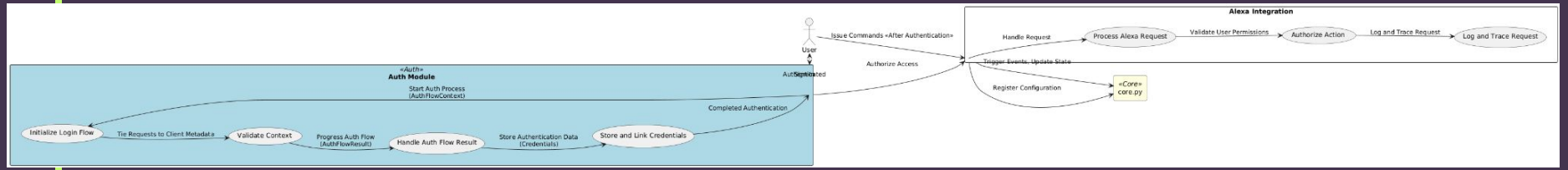
Concrete

Components Cont.

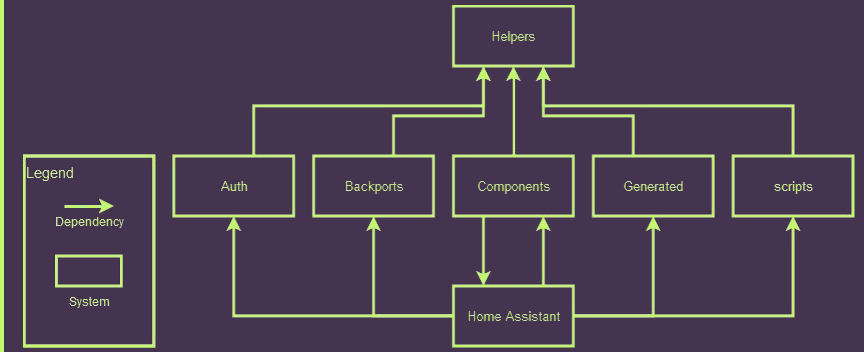
- User Imports
- User imports at auth module
- User decorators, commands, and connections imports



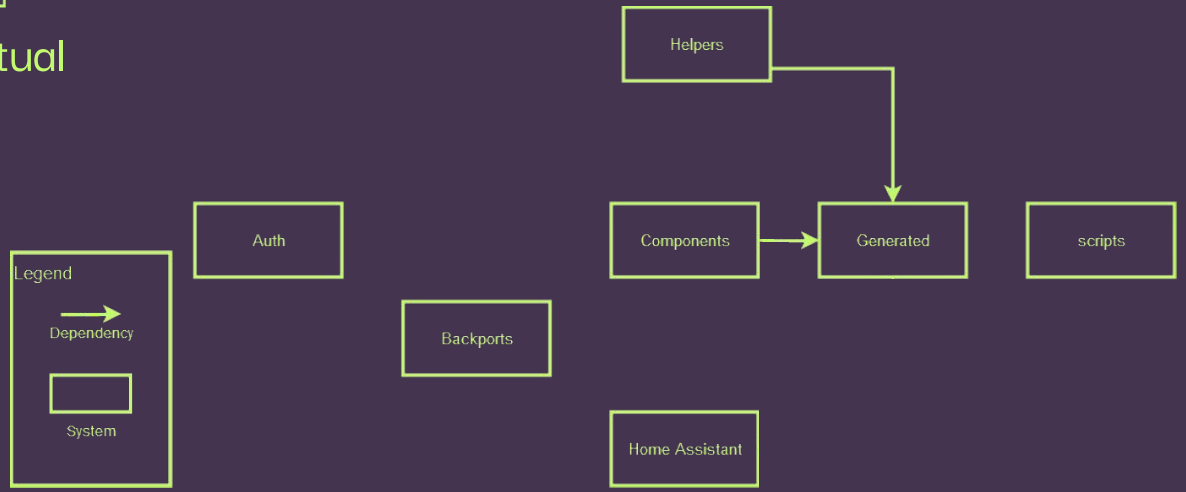
Use Case of Auth



Divergences Of Generated



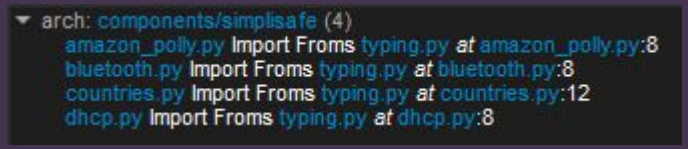
Conceptual



Concrete

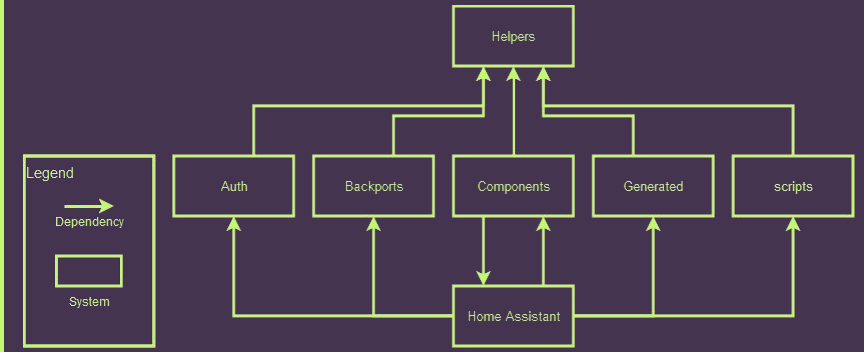
Generated Cont.

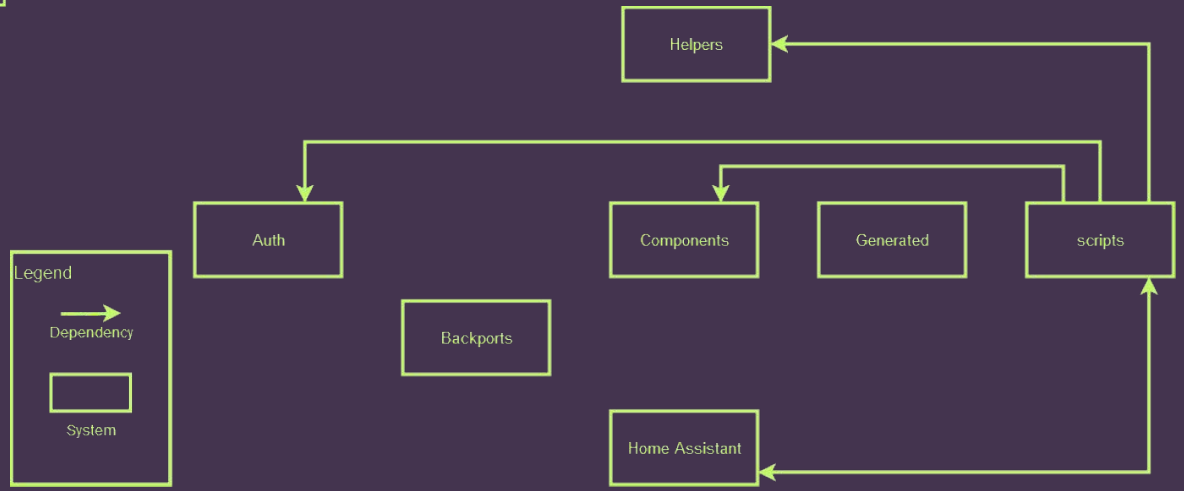
- Only four dependencies
- All important from typing.py from various sub components
- Final is defined more specifically in the code later



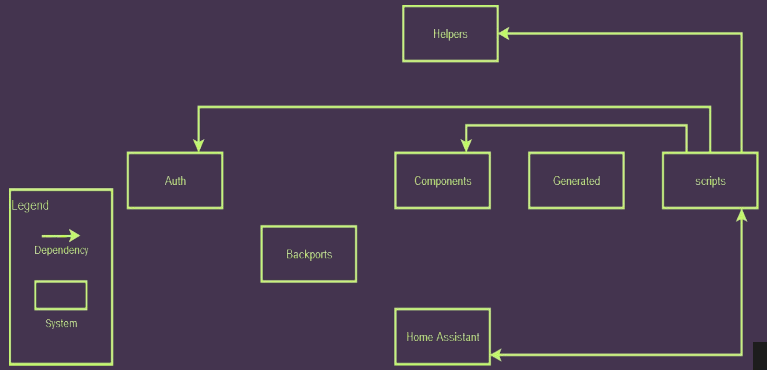

```python
from __future__ import annotations

from typing import Final

SUPPORTED_ENGINES: Final[set[str]] = {
    "generative",
    "long-form",
    "neural",
    "standard",
}
```

Divergences Of Scripts





Scripts Cont.




```
from typing import TYPE_CHECKING

from homeassistant import runner
from homeassistant.auth import auth_manager_from_config
from homeassistant.auth.providers import homeassistant as hass_auth
from homeassistant.config import get_default_config_dir
from homeassistant.core import HomeAssistant
from homeassistant.helpers import device_registry as dr, entity_registry as er
```

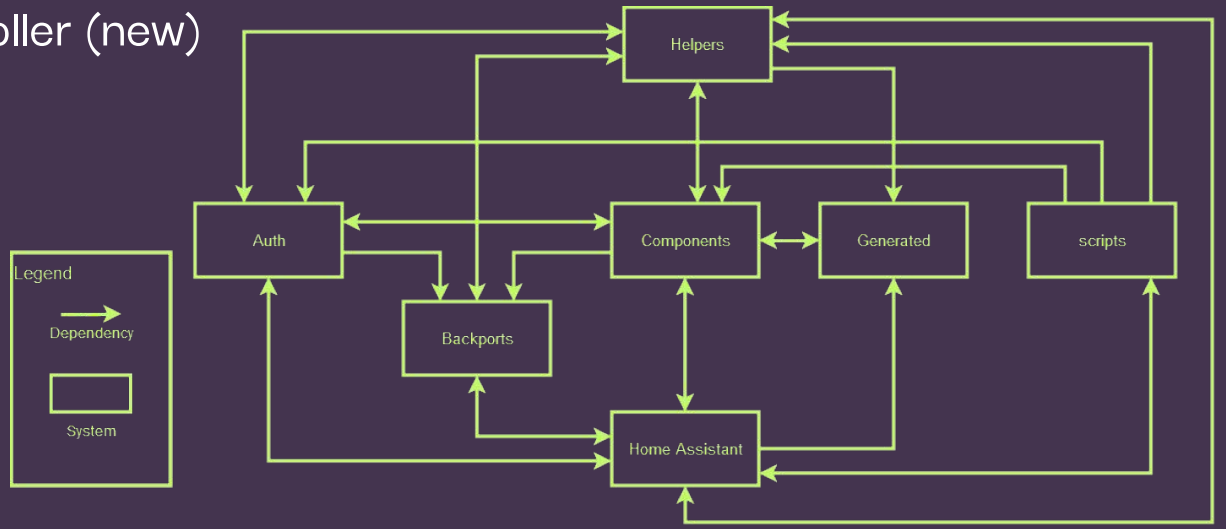
```
def main() -> int:
    """Start Home Assistant."""
    validate_python()

    args = get_arguments()

    if not args.ignore_os_check:
        validate_os()

    if args.script is not None:
        # pylint: disable-next=import-outside-toplevel
        from . import scripts

        return scripts.run(args.script)
```

03

Lessons Learnt

Overall Architecture

- Event-Bus
- Microservices
- Model - View - Controller (new)



Limitations of findings

- Open Source with almost 4000 contributors
    - Jason Hu committed in summer 2018 and winter 2019, not to be seen again
    - No real roles except for Paulus
- Architectures diagrams found online are based off the Concrete

Lessons Learned

- For some functions, proper comments are provided to explain the function but in most cases, the comments are reserved to giving a short explanation of the code's function and explain a line.
- This makes understanding the code very difficult as we would have to analyze the code and figure out or infer what a line/function does.
- This struggle showed us the importance of proper documentation and how it helps with understanding the architecture and it helps with future maintenance.

THANK YOU