PROPOSED FILESPEC SPLIT OUT:

https://docs.google.com/document/d/1_mrxQpqnn7vbkdLrMQxc7nBYLFnCenc06TcjXSOGKdg/edit#heading=h.6too4ygvhex1

OpenCompose mapping to Docker Compose, Kubernetes & Ansible Compose

OpenCompose Objects Mappings

Object	Description	Ansible Compose / Docker Compose Object	Kubernetes Object
build	Builds a container using a linux container	build	**BuildConfig**
context	Either a path to a directory	context	**BuildConfig**.spec.source.conte

	containing a *Dockerfile*, or a url to a git repository		xtDir
dockerfile	Alternate *Dockerfile*.	dockerfile	N/A - This is the default file used by the **BuildConfig**/Docker strategy
command	Override the default command.	command	**Pod**.spec.container.command
container_name	Specify a custom container name, rather than a generated default name.	container_name	**ObjectMeta**/name
depends_on	Express dependency between services	depends_on	Directed service dependencies (coming soon)
environment	Add environment variables	environment	**Container/**env
expose	Expose ports without publishing them to the host machine	expose	**Service**
extends	Extend another service, in the current file or another, optionally overriding configuration.	extends	N/A
image	Specify the image to start the container from.	image	**BuildConfig**/image
labels	Add metadata to containers using Docker labels	labels	**ObjectMeta**/annotations

ports	Expose ports.	ports	**Service**[1]
volumes	Mount paths or named volumes	volumes	**DeploymentConfig**/volume[2]
volumes_from	Mount all of the volumes from another service or container	volumes_from	N/A - use volumes instead Colocation + volume mounts of all the defined mounts on the first image.
entrypoint	Override the default entrypoint.	entrypoint	**Pod**.spec.container.command
env_file	Add environment variables from a file. Can be a single value or a list.	env_file	N/A
links	Link to containers in another service. Either specify both the service name and a link alias (SERVICE:ALIAS), or just the service name.	links	Pod.spec.container

Mapping exists	Mapping not required	Problematic

Open Questions

[1] A non-trivial number of docker compose files support remapping ports - and so pod -> service relationships involve three entities (pod port, service port -> pod port, and service external port)

[2] Volumes, as defined in a compose file, can express implicit colocation into a pod. So the act of sharing a volume mount is == being in a pod.

How do we allow the developer to express that services should be colocated? Comments: Network info / Volume Sharing / links is currently used to express this in Docker compose.
How do we allow the developer express minimum capabilities required to run (e.g. disk space, memory, CPU)?
How do we compose these compose files?

Non-Relevant docker-compose configurations

The following docker-compose configuration options are listed here but are **not often used by developers.**

Docker-compose configuration	Description
cap_add, cap_drop	Add or drop container capabilities
cgroup_parent	Specify an optional parent cgroup for the container
devices	List of device mappings. Uses the same format as the --device docker client create option
dns	Custom DNS servers. Can be a single value or a list
dns_search	Custom DNS search domains. Can be a single value or a list.
tmpfs	Mount a temporary file system inside the container. Can be a single value or a list.

external_links	Link to containers started outside this docker-compose.yml or even outside of Compose, especially for containers that provide shared or common services.
extra_hosts	Add hostname mappings. Use the same values as the docker client --add-host parameter.
logging	Logging configuration for the service.
log_driver, log_driver, log_opt	
network_mode	Network mode. Use the same values as the docker client --net parameter, plus the special form service:[service name].
networks	Networks to join, referencing entries under the top-level networks key.
aliases	Aliases (alternative hostnames) for this service on the network.
ipv4_address, ipv6_address	Specify a static IP address for containers for this service when joining the network.
pid	Sets the PID mode to the host PID mode.
security_opt	Override the default labeling scheme for each container.
stop_signal	Sets an alternative signal to stop the container.

ulimits	Override the default ulimits for a container. You can either specify a single limit as an integer or soft/hard limits as a mapping
volume_driver	
cpu_shares, cpu_quota, cpuset, domainname, hostname, ipc, mac_address, mem_limit, memswap_limit, privileged, read_only, restart, shm_size, stdin_open, tty, user, working_dir	Each of these is a single value, analogous to its docker run counterpart.