

- local SAFETY: will not outlive its region, may be stack allocated
- local REQUIREMENT: do not escape region
- local parameter isn't a requirement, it's a behavior guarantee
- global function can't capture local values
- No heap to stack pointer
- XING: Applies only to non immediate data

Uniqueness <i>past</i> <div>unique < aliased</div> <ul style="list-style-type: none"> • unique LINEAGE: has not been aliased • unique ENTITLEMENT: may be overwritten • XING: Applies only to mutable and mutable containing data 	<div>many < once</div> <div>Affinity <i>future</i></div> <ul style="list-style-type: none"> • once SAFETY: function will not create an alias • once REQUIREMENT: zero or one call allowed • many function can't capture unique values • Function capturing unique is once • XING: Applies only to functions and function containers
<div>uncontended < shared < contended</div> <div>Contention <i>past</i></div> <ul style="list-style-type: none"> • Non contended LINEAGE: value retains access lock • ENTITLEMENT: <ul style="list-style-type: none"> – uncontended: may be read or written – share: may only be read • XING: Applies only to mutable and mutable containing data 	<div>portable < nonportable</div> <div>Portability <i>future</i></div> <ul style="list-style-type: none"> • nonportable SAFETY: function will not break captured non contended lineage • nonportable REQUIREMENT: ... • portable function can't capture non contended value • Function capturing non contended is nonportable • XING: Applies only to functions and function containers

unyielding < yielding

Yielding
future

- yielding SAFETY: function will not perform effect handled in parent stack
- yielding REQUIREMENT: ...
- XING: Applies only to functions and function containers

<div>read_write < read < immutable</div> <div>Visibility <i>past</i></div> <ul style="list-style-type: none"> • Non immutable LINEAGE: value ... • ENTITLEMENT: <ul style="list-style-type: none"> – read_write: may be read or written – read: may only be read • XING: Applies only to mutable and mutable containing data 	<div>stateless < observing < stateful</div> <div>Statefulness <i>future</i></div> <ul style="list-style-type: none"> • SAFETY: <ul style="list-style-type: none"> – observing: will not W on captured read_write data – stateful: will not read or write over captured non immutable data • REQUIREMENT: <ul style="list-style-type: none"> – observing: no write – stateful: value ... • XING: Applies only to functions and function containers
---	---