On actions - Hearts of Iron 4 Wiki

This is a community maintained wiki. If you spot a mistake then you are welcome to fix it.

On actions are blocks that are executed when a certain action occurs, such as a country declaring war on a different country or a state changing control. On actions are stored in /Hearts of Iron IV/common/on_actions/*.txt files.

Each on action is a separate block within the `on_actions = { ... }` block. Each on_action has up to 2 arguments:

`effect = { ... }` is present for every single on action, being an effect block the insides of which are executed when needed.

`random_events = { ... }` is present for on_actions where the default scope (Same as ROOT, if not specified otherwise) is a country, such as on_new_term_election. This instantly fires a random one of the specified events within with the given weights being applied. This is done with a ==probability-proportional-to-size sampling== approach.

> Putting `0` instead of an event ID will ensure that nothing will happen if the chance lands on this.
> Addiitonally, an event cannot be fired using `random_events = { ... }` if the event's `trigger = { ... }` block evaluates as false. In this case, each scope is treated the same as on action's: on action's FROM is treated as the event's FROM, same with FROM.FROM.
> If there are multiple random_events blocks, one event will be picked from each.

Note that in terms of Scopes, ROOT is the default assumed scope unless specified otherwise (in on_actions that have THIS as a separate entity from ROOT), while FROM and FROM.FROM can serve as secondary blocks that are provided in addition.

File example[编辑 | 编辑源代码]

```
on_actions = {
  on_startup = {
    effect = { # NEVER FORGET! Important to include this line to distinguish it from random_events = { ... }
      every_country = {
        limit = {
          is_ai = no
        }
        country_event = welcome_event.1
      }
      ENG = {
        country_event = {
          id = new_year.1
          days = 365  # Fires on January 1 1937. Remember that leap days do not exist in-game.
        }
      }
    }
  }
  on_state_control_changed = {
    random_events = {
      1 = germany_state_control.1 # Assuming the triggers for the events are met, then
      1 = germany_state_control.2 # fires one of germany_state_control.1 or germany_state_control.2
      3 = 0                # Each has a 20% chance, and there's 60% chance nothing happens.
    }
    effect = {
      if = {
        limit = {   # Execute if Italy captures Corsica or Savoy from France
          tag = ITA
          FROM = { tag = FRA }
          FROM.FROM = {
            OR = {
              state = 1
              state = 735
            }
          }
        }
        FROM.FROM = {
          set_resistance = 60
          damage_building = {
            type = infrastructure
            damage = 2
          }
        }
      }
    }
  }
}
```

General on actions[编辑 | 编辑源代码]

Name	Description	Examples	Notes	Version Added
on_startup	Trigger the following commands at the first day of a new game (doesn't work with save loading).	`on_startup = { effect = { … } }`	**Does not have a default scope**, rather than being fired for every country. Each country must be scoped into.	1.3.3
on_daily	Triggers each day for every country separately (performance heavy, use carefully)	`on_daily = { effect = { … } }`	Useful for scripted guis and mods adding new mechanics (can increment a variable daily e.g.).	1.5.2
on_daily_TAG	Triggers each day for the specified country only	`on_daily_SOV = { effect = { … } }`	Only runs the effects if the country exists.	1.9
on_weekly	Triggers each week for every country separately	`on_weekly = { effect = { … } }`	Useful for ai scripting.	1.9

Name	Description	Examples	Notes	Version Added
on_weekly_TAG	Triggers each week for the specified country only	`on_weekly_GER = { effect = { … } }`	Only runs the effects if the country exists.	1.9
on_monthly	Triggers each month for every country separately	`on_monthly = { effect = { … } }`		1.9
on_monthly_TAG	Triggers each month for the specified country only	`on_monthly_USA = { effect = { … } }`	Only runs the effects if the country exists.	1.9
on_nuke_drop	Trigger the following commands whenever a country drops a nuke.	`on_nuke_drop = { effect = { set_global_flag = first_nuke_dropped } }`	FROM is the nuked state.	1.0
on_pride_of_the_fleet_sunk	Triggers when a country's pride of the fleet is sunk	`on_pride_of_the_fleet_sunk = { effect = { … } }`	FROM is the killer country, ROOT is the country of that lost its pride of the fleet.	1.6
on_naval_invasion	Triggers the following commands whenever a sea invasion is made.	`on_naval_invasion = { effect = { … } }`	THIS state that is invaded, ROOT country that invades, FROM state that the invasion started	1.9
on_paradrop	Triggers the following commands whenever a landing occurs.	`on_paradrop = { effect = { … } }`	THIS state that is invaded, ROOT country that invades, FROM state that the invasion started	1.9

Politics[编辑 | 编辑源代码]

Name	Description	Examples	Notes	Version Added

on_coup_succeeded	Trigger the following commands whenever a coup succeeds.	`on_coup_succeeded = {`
		`    effect = {`
		`        random_other_country = {`
		`            limit = {`
		`                has_government = democratic`
		`                original_tag = ROOT`
		`            }`
		`            set_politics = { elections_allowed = yes }`
		`        }`
		`    }`
		`}`
				1.0

on_government_change	Trigger the following commands whenever a country switch government.	`on_government_change = { effect = { … } }`		1.0
on_ruling_party_change	Trigger the following commands whenever a country switch ideology.	`on_ruling_party_change = { effect = { … } }`		1.9
on_new_term_election	Trigger the following commands whenever an election happen or is called by the **hold_election** command.	`on_new_term_election = { random_events = { 100 = usa.6 } }`		1.0
on_peaceconference_ended	Trigger the following commands whenever a peace conference ends.	`on_peaceconference_ended = { effect = { … } }`	ROOT is the winner, FROM is the loser.	1.5
on_peaceconference_started	Trigger the following commands whenever a peace conference starts.	`on_peaceconference_started = { effect = { … } }`	ROOT is the winner, FROM is the loser.	1.12.3

Diplomacy/War[编辑 | 编辑源代码]

Name	Description	Examples	Notes	Version Added

on_send_volunteers	Trigger the following commands whenever a country send volunteers to another.	`on_send_volunteers = {`
		`    effect = { ... }`
		`}`
			ROOT is sender, FROM is receiver.	1.9

on_recall_volunteers	Trigger the following commands whenever a country recalls volunteers from another.	`on_recall_volunteers = {`
		`    effect = { ... }`
		`}`
			ROOT is recaller, FROM is originator.	unknown - 1.9

on_border_war_lost	Trigger the following commands whenever a country lose a border war.	`on_border_war_lost = {`
		`    effect = {`
		`        owner = {`
		`            add_ideas = lost_conflict`
		`        }`
		`    }`
		`}`
				1.0

on_war_relation_added	fired when two countries end up at war with each other (on_war is fired when a country goes to war against anyone and is not fired again when it enters war against another country unless it went to peace first)	`on_war_relation_added = {`
		`    effect = { ... }`
		`}`
			ROOT is attacker, FROM is defender	1.9.3

on_declare_war	Trigger the following commands whenever a country declares war.	`on_declare_war = {`
		`    effect = {`
		`        if = {`
		`            limit = {`
		`                tag = GER`
		`                FROM = { tag = SOV }`
		`            }`
		`            add_ideas = GER_barbarossa`
		`        }`
		`    }`
		`}`
			FROM is war target.
			ROOT is for the country who is declaring war
				1.0

on_war	Trigger the following commands whenever a country has just entered a state of war from initially being at peace.	`on_war = {`
		`    effect = { ... }`
		`}`
			THIS is country that has just gotten into a war.	1.7

on_peace	Trigger the following commands whenever a country is no longer at war.	`on_peace = {`
		`    effect = { ... }`
		`}`
			THIS is country that is no longer at war.	1.7

on_capitulation	Trigger the following commands whenever a country capitulates, in the middle of the process.	`on_capitulation = {`
		`    effect = { ... }`
		`}`
			ROOT is capitulated country, FROM is winner. Several processes such as the deletion of units and transfer of equipment have already been executed by this point.	1.0

on_capitulation_immediate	Trigger the following commands whenever a country capitulates, at the beginning of the process.	`on_capitulation_immediate = {`
		`    effect = { ... }`
		`}`
			ROOT is capitulated country, FROM is winner.	1.11.5

Name	Description	Examples	Notes	Version Added

on_uncapitulation	Trigger the following commands whenever a country that was previously capitulated changes its status to no longer having capitulated.	on_uncapitulation = {
		    effect = { ... }
		}
			ROOT is the country affected.	1.4

on_annex	Trigger the following commands whenever a country is annexed.	on_annex = {
		    effect = { ... }
		}
			ROOT is winner, FROM gets annexed. For civil wars **on_civil_war_end** is also fired.	1.3.3

on_civil_war_end_before_annexation	Trigger the following commands just before FROM gets annexed, meaning the country and everything it owns still exists.	on_civil_war_end_before_annexation = {
		    effect = { ... }
		}
			ROOT is winner, FROM gets annexed. It will also fire **on_annex** and **on_civil_war_end**.	1.6

on_civil_war_end	Trigger the following commands whenever a civil war ends.	on_civil_war_end = {
		    effect = { ... }
		}
			ROOT is civil war winner, FROM gets annexed. This will also fire **on_annex**.	1.0

on_puppet	Trigger the following commands whenever a country is puppeted in a **peace conference only**.	on_puppet = {
		    effect = { ... }
		}
			ROOT is the nation being puppeted, FROM is the overlord.	1.0

on_liberate	Trigger the following commands whenever a country is liberated in a **peace conference only**.	on_liberate = {
		    effect = { ... }
		}
			ROOT is the nation being liberated, FROM is the leader of the liberators.	1.0

on_release_as_free	Trigger the following commands whenever a country is released.	on_release_as_free = {
		    effect = { ... }
		}
			#ROOT is free nation FROM is releaser.	1.3

on_release_as_puppet	Trigger the following commands whenever puppeting through the occupied territories menu during peace time (or when releasing from non-core but owned territory).	on_release_as_puppet = {
		    effect = { ... }
		}
			ROOT is the nation being released, FROM is the overlord.	1.3

Faction[编辑 | 编辑源代码]

Name	Description	Examples	Notes	Version Added
on_create_faction	Trigger the following commands whenever a country create a faction.	on_create_faction = { effect = { … } }	FROM is the one that joins the faction.	1.0
on_faction_formed	Trigger the following commands when a faction is formed.	on_faction_formed = { effect = { news_event = { id = news.159 } } }		1.0
on_offer_join_faction	Trigger the following commands whenever a country is invited in a faction.	on_offer_join_faction = { effect = { … } }	FROM is the country invited.	1.0
on_join_faction	Trigger the following commands for a faction leader whenever a country is invited in his faction.	on_join_faction = { effect = { … } }	FROM is faction leader on join faction requests.	1.0
on_assume_faction_leadership	Trigger the following commands whenever a country assumes leadership of a faction.	on_assume_faction_leadership = { effect = { … } }	FROM is the former faction leader.	1.7
on_leave_faction	Trigger the following commands whenever a country leave a faction.	on_leave_faction = { effect = { if = { limit = { AND = { tag = CAN NOT = { has_dlc = "Together for Victory" } } } drop_cosmetic_tag = yes } }		1.0

Autonomy[编辑 | 编辑源代码]

Name	Description	Examples	Notes	Version Added
on_subject_annexed	Trigger the following commands when a country annex a subject.	on_subject_annexed = { effect = { … } }	ROOT is the subject, FROM is the overlord.	1.0
on_subject_free	Trigger the following commands when a country grants freedom to a puppet.	on_subject_free = { effect = { … } }	ROOT is the subject, FROM is the previous overlord.	1.0
on_subject_autonomy_level_change	Trigger the following commands when the autonomy level of a puppet changes.	on_subject_autonomy_level_change = { effect = { … } }	ROOT is the subject, FROM is the overlord.	1.0

Governments in Exile[编辑 | 编辑源代码]

Name	Description	Examples	Notes	Version Added
on_government_exiled	Trigger the following commands whenever a country becomes a government in exile.	on_government_exiled = { effect = { = { … } }	ROOT is the government in exile, FROM is the country that is hosting the government in exile.	1.6
on_host_changed_from_capitulation	Trigger the following commands whenever a country that is hosting a government in exile has capitulated.	on_host_changed_from_capitulation= { effect = { = { … } }	ROOT is the government in exile, FROM is the new country hosting the government in exile, FROM:FROM is the old country that was hosting the government in exile.	1.6
on_exile_government_reinstated	Trigger the following commands whenever a country has returned from governing in exile.	on_exile_government_reinstated = { effect = { = { … } }	ROOT is the government in exile, FROM is the country that was hosting the government in exile.	1.6

States[编辑 | 编辑源代码]

Name	Description	Examples	Notes	Version Added
on_state_control_changed	Trigger the following commands when a state's controller changes.	on_state_control_changed = { effect = { … } }	ROOT is new controller, FROM is old controller, FROM.FROM is state ID.	1.4

Wargoals[编辑 | 编辑源代码]

Name	Description	Examples	Notes	Version Added
on_justifying_wargoal_pulse	Trigger the following commands whenever the country is targeted by a wargoal under justification.	on_justifying_wargoal_pulse = { random_events = { 100 = war_justification.1 } }	FROM = target nation. Checked every day.	1.0

Name	Description	Examples	Notes	Version Added
on_wargoal_expire	Trigger the following commands whenever a wargoal expire.	on_wargoal_expire = { random_events = { 100 = war_justification.301 } }	FROM is the wargoal owner.	1.0

Unit Leader[编辑 | 编辑源代码]

Name	Description	Examples	Notes	Version Added
on_unit_leader_created	Trigger the following commands when an army leader is created.	on_unit_leader_created = { effect = { … } }	FROM is owner country, ROOT is the unit leader.	1.5
on_army_leader_daily	Trigger the following commands on an army leader each day.	on_army_leader_daily = { effect = { … } }	FROM is owner country, ROOT is the unit leader.	1.0
on_army_leader_won_combat	Trigger the following commands whenever an army leader won a combat.	on_army_leader_won_combat = { effect = { … } }	FROM is owner country, ROOT is the unit leader.	1.0
on_army_leader_lost_combat	Trigger the following commands whenever an army leader lost a combat.	on_army_leader_lost_combat = { effect = { … } }	FROM is owner country, ROOT is the unit leader.	1.0
on_unit_leader_level_up	Trigger the following commands when a leader gain a level.	on_unit_leader_level_up = { effect = { … } }	FROM is owner country, ROOT is the unit leader.	1.0
on_army_leader_promoted	Trigger the following commands whenever a corps commander is promoted to a field marshal.	on_army_leader_promoted = { effect = { add_timed_unit_leader_trait = { trait = recently_promoted days = 100 } } }	FROM is owner country, ROOT is the unit leader.	1.0
on_unit_leader_promote_from_ranks_veteran	Triggers the following commands whenever a unit commander gets promoted to a general.	on_unit_leader_promote_from_ranks_veteran = { effect = { … } }	FROM is unit, OWNER is owner country, ROOT is the unit leader.	1.12
on_unit_leader_promote_from_ranks_green	Triggers the following commands whenever an unit commander gets promoted to a general.	on_unit_leader_promote_from_ranks_green = { effect = { … } }	FROM is unit, OWNER is owner country, ROOT is the unit leader.	1.12

Military[编辑 | 编辑源代码]

Name	Description	Examples	Notes	Version Added
on_add_history	Triggers the following commands whenever receiving a history entry.	on_add_history = { effect = { … } }	ROOT is the unit.	1.12

Aces[编辑 | 编辑源代码]

Name	Description	Examples	Notes	Version Added
on_ace_promoted	Trigger the following commands whenever an ace is created.	on_ace_promoted = { random_events = { 100 = ace_promoted.1 } }	FROM = ace.	1.0
on_ace_killed	Trigger the following commands whenever an aces is killed.	on_ace_killed = { random_events = { 100 = ace_died.1 } }	FROM = ace.	1.0
on_ace_killed_on_accident	Trigger the following commands whenever our aces died on accident.	on_ace_killed_on_accident = { random_events = { 100 = ace_died.1 } }	FROM = our ace died in accident.	1.9
on_non_ace_killed_other_ace	Trigger the following commands whenever non ace killed enemy ace.	on_non_ace_killed_other_ace = { FROM = { random_events = { 100 = ace_died.1 } } }	FROM = enemy ace.	1.9
on_ace_killed_by_ace	Trigger the following commands whenever an aces is killed by another ace.	on_ace_killed_by_ace = { random_events = { 100 = ace_killed_by_ace.1 } }	FROM = our ace, PREV = enemy ace, has killed FROM.	1.0
on_ace_killed_other_ace	Trigger the following commands whenever an aces is killed by another ace (surviving ace side).	on_ace_killed_other_ace = { random_events = { 100 = ace_killed_other_ace.1 } }	FROM = our ace, PREV = enemy ace, killed by FROM.	1.0
on_aces_killed_each_other	Trigger the following commands whenever two aces kill each other in air duel.	on_aces_killed_each_other = { random_events = { 100 = aces_killed_each_other.1 } }	FROM = ace, PREV = enemy ace.	1.0

La Résistance[编辑 | 编辑源代码]

Name	Description	Examples	Notes	Version Added
on_operation_completed	Trigger the following commands whenever an operative completed.	on_operation_completed = { effect = { … } }	THIS - the operative, ROOT - the initiating country, FROM - the target country.	1.9
on_operative_detected_during_operation	Trigger the following commands whenever an operative dies.	on_operative_death = { effect = { … } }	THIS - the operative, ROOT - the killer country (optional), FROM - the country the operative is operating for, FROM.FROM - operation state (will only be set if the operation has a specific selection_target).	1.9
on_operative_on_mission_spotted	Trigger the following commands whenever an operative performing an offensive mission in a country.	on_operative_on_mission_spotted = { effect = { … } }	THIS - the operative, FROM - the country the operative was performing its mission in, ROOT - the country the operative is operating for.	1.9
on_operative_captured	Trigger the following commands whenever an operative is captured.	on_operative_captured = { effect = { … } }	THIS - the operative, ROOT - the country the operative was performing its mission in, FROM - the country the operative is operating for.	1.9
on_operative_created	Trigger the following commands whenever an operative is created.	on_operative_created = { effect = { … } }	THIS - the operative, FROM - the country the operative is created by.	1.9.1
on_operative_death	Trigger the following commands whenever an operative dies.	on_operative_death = { effect = { … } }	THIS - the operative, ROOT - the killer country (optional), FROM - the country the operative is operating for.	1.9
on_operative_recruited	Trigger the following commands whenever an operative is recruited.	on_operative_recruited = { effect = { … } }	THIS - the operative, FROM - the country the operative is created by.	1.9.1
on_fully_decrypted_cipher	Trigger the following commands whenever a country fully decrypts cipher of a target country.	on_fully_decrypted_cipher = { effect = { … } }	THIS - the target country that its cipher is decrypted, FROM - the decrypter country.	1.9
on_activated_active_decryption_bonuses	Trigger the following commands whenever a country activates its active cipher bonuses against a target.	on_activated_active_decryption_bonuses = { effect = { … } }	THIS - the target country, FROM - the country that activates its bonuses.	1.9