Olympic Data Feed

ODF Cycling Mountain Bike Data Dictionary for the XX Commonwealth Games

16 June 2013
Technology and Information Department

INTERNATIONAL OLYMPIC COMMITTEE
Château de Vidy, 1007 Lausanne, Switzerland, Tel. +41 21 621 6111 / Fax +41 21 621 6216 / www.olympic.org

TABLE OF CONTENT

1 Introduction

1.1 This document

This document is a Derivative Work (as defined in the License hereto) prepared by Glasgow 2014 Limited for the purpose of the XX Commonwealth Games.

1.2 Objective

The objective of this document is to provide a formal definition of the ODF Cycling Mountain Bike Data Dictionary for the XX Commonwealth Games, with the intention that the information message producer and the message consumer can successfully interchange the information as the Cycling Mountain Bike competition is run.

1.3 Main Audience

The main audience of this document is the IOC as the ODF promoter, Glasgow 2014, ODF users such as the World News Press Agencies, Rights Holding Broadcasters and International Sports Federations.

1.4 Glossary

The following abbreviations are used in this document

Acronym	Description
IF or International Federation	The international governing body of an Olympic Sport as recognized by the IOC
IOC	International Olympic Committee
IPC	International Paralympic Committee
CGA	Commonwealth Games Associations
ODF	Olympic Data Feed
ODF-PiT	Olympic Data Feed Point in Time, messages that are generated at certain point during competition
ODF-RT	Olympic Data Feed Real Time, messages that are generated when available
RSC	Results System Codes, determine uniquely one unit of the competition, specifying the discipline, gender, event, phase and unit.
Sport	is administered by an international federation and can be composed of one or more disciplines
WNPA	World News Press Agencies

1.5 Related Documents

Document Reference	Document Title	Document Description
ODF/INT001	ODF Message Transmission Document	This document describes the technical standards to be used to transfer ODF messages

		between the message generators and the final ODF users
ODF/COD001	ODF Common Codes Document	This document describes the ODF codes used across the rest of the ODF documents
ODF/INT142	ODF General Messages Interface Document	This document describes the ODF general messages for the XX Commonwealth Games

2 Overall Perspective

2.1 Objective

The objective of this document is to focus on the formal definition of the ODF Cycling Mountain Bike Data Dictionary.

2.2 End to End data flow

In the following sections, for each ODF General message it will be explained in further detail those elements, attributes, codes, ODF header, the trigger and frequency for each message generation, as well as the sort of the message that are particular in the case of Cycling Mountain Bike.

Any ODF Cycling Mountain Bike message should follow all the previous definitions in order to be considered as an ODF compliant message.

3 Codes

Several codes are used in the definition of the messages in this document. Any code will be referenced the following way:

CC @CodeEntity

CodeEntity is the name of the entity that identifies a particular set of codes.

The following table describes the codes entities used in document sorted by name, indicating whether the set of values can be found in the ODF Common Codes Document, or listed in the table itself, otherwise. Please refer to ODF General Messages Interface Document to know the format of these codes.

Code Entity	**Code Entity Set of Values**	
CC @IntPtType	**Code**	**Description**
	FL	Finish Loop
	LAP	Lap
	HL	Half Lap
	SL	Start Loop
CC @IRM	**Code**	**Description**
	DNF	Did not finish
	DNS	Did not start
	DSQ	Disqualified
	LAP	Lapped
CC @ResultType	**Code**	**Description**
	IRM	For IRM status
	RANK	Rank without final result time
	TIME	Time
CC @TemperatureUnit	**Code**	**Description**
	C	Celsius

4 Point in Time

4.1 Point in Time Applicable Messages

The following table is a full list of all ODF messages and describes the list of messages used in Cycling Mountain Bike, as well as the category of each message, which identifies if the message structure definition can be found in the ODF General Messages Interface Document.

- The column “Message type” indicates the DocumentType that identifies a message
- The column “Message name” is the message name identified by the message type
- The column “Message used in this sport” indicates whether a message is used in particular for this sport or not. If it is not ticked (X), then the message should not be used for this sport.
- The column “Message extended in this document” indicates whether a particular message has extended definition in regards to those that are general for all sports. Any message ticked (X) in this column should also be ticked in the “Message used in this sport column”. If one message has extended definition, it should be considered both, the extensions as well as the general rules for one message that is used in the case of the sport. However, if one particular message is not extended, then it should follow the general definition rules.

Message Type	Message name	Message used in this sport	Message extended in this document
DT_SCHEDULE	Competition schedule	X	
DT_SCHEDULE_UPDATE	Competition schedule update	X	
DT_PARTIC	List of participants by discipline	X	X
DT_PARTIC_UPDATE	List of participants by discipline update	X	X
DT_PARTIC_TEAMS	List of teams		
DT_PARTIC_TEAMS_UPDATE	List of teams update		
DT_MEDALS	Medal standings	Global	
DT_MEDALLISTS_DAY	Medallists of the day	Global	
DT_HISTORIC_RECORD	Historical records		
DT_GLOBAL_GM	Global good morning	Global	
DT_GLOBAL_GN	Global good night	Global	
DT_START_LIST	Start List	X	X
DT_RESULT	Event Unit Results	X	X
DT_PHASE_RESULT	Phase Results		
DT_CUMULATIVE_RESULT	Cumulative Results		
DT_POOL_STANDING	Pool Standings of group in a team competition		

DT_RANKING	Event Final ranking	X	
DT_STATS	Statistics table		
DT_MEDALLISTS	Medallists of one event	X	
DT_MEDALLISTS_DISCIPLINE	Medallists by discipline	X	
DT_RECORD	Records		
DT_COMMUNICATION	Official Communication	X	X
DT_BRACKETS	Brackets		
DT_GM	Discipline/venue good morning	X	
DT_GN	Discipline/venue good night	X	
DT_FED_RANKING	Federation Ranking		
DT_CONFIG	Discipline configuration	X	X
DT_WEATHER	Event Unit Weather conditions	X	X
DT_SERIAL	List of Current PiT Serial	X	

4.1.1 List of participants by discipline/ List of participants by discipline update

4.1.1.1 Description

This message is the List of participants by discipline (and the update), for that discipline it is the list of athletes, as described in the ODF General Messages Interface Document.

4.1.1.2 Header Values

The definition in the ODF General Messages Interface Document is valid

4.1.1.3 Trigger and Frequency

Please, follow the general definition.

4.1.1.4 Message Structure

The optional elements defined for this message in the ODF General Messages Interface Document that should be included in the case of Cycling Mountain Bike are:

- EventEntry

In the next section (message values), there is a more detailed definition.

4.1.1.5 Message Values

The following table lists the "List of participants by discipline/ update" optional attributes (defined in the ODF General Messages Interface Document) that are used in the case of Cycling Mountain Bike, as well as the attributes that have an extended definition.

Element	Attribute	M/O	Value	Comments
Participant	GivenName	M	S(25)	Given name in WNPA format (mixed case)
	BirthDate	O	YYYYMMDD	Date of birth
	Height	O	N(3) 999	Height in centimetres Send when this information is available
	Weight	O	N(3) 999	Weight in kilograms Send when this information is available
	MainFunctionId	M	CC @Function	Main function
Discipline	InternationalFederationId	O	S(16)	UCI code (competitor's federation number for the discipline). It will be included.
RegisteredEvent	Bib	O	S(2)	Bib number. Although this attribute is optional, it will be updated and informed as soon as this information is known. Example: 8, 10 …

The following table describes in more detail the EventEntry element in the case of Cycling Mountain Bike.

Element: EventEntry			
Type	**Code**	**Value**	**Description**
E_ENTRY	E_SUBSTITUTE	S(1)	For @Type: Send proposed type
			For @Code: Send proposed code
			For @Value: Send "Y" if the competitor is a substitute or N if it is not more.
	~~E_RANK~~	~~S(4)~~	~~For @Type:~~

Element: EventEntry			
Type	**Code**	**Value**	**Description**
			~~Send proposed type~~
			~~For @Code:~~ ~~Send proposed code~~
			~~For @Value:~~ ~~Send the UCI ranking for the competitor.~~

For the table above, we have the following additional/summary information:

Type /Code	Description	Expected
E_ENTRY /E_SUBSTITUTE	Flag that indicates the competitor is a substitute.	As soon as this information is (this information can be sent in both messages)
~~E_ENTRY /E_RANK~~	~~UCI Ranking for the competitor.~~	~~As soon as the venue results has this information~~ ~~(this information can be sent in both messages)~~

4.1.1.6 Message sort

Please, follow the general definition.

4.1.2 Start List

4.1.2.1 Description

This message is the Start List message as described in the ODF General Messages Interface Document.

4.1.2.2 Header Values

The DocumentCode attribute in the ODF header will be sent according to the ODF Common Codes document (header values sheet).

4.1.2.3 Trigger and Frequency

Please, follow the general definition.

4.1.2.4 Message Structure

The optional elements defined for this message in the ODF General Messages Interface Document that should be included in the case of Cycling Mountain Bike are:

- UnitDateTime (following the general rules for this element)
- UnitInfo

4.1.2.5 Message Values

The following table lists the Start List optional attributes (defined in the ODF General Messages Interface Document) that are used in the case of Cycling Mountain Bike, as well as the attributes that have an extended definition.

Element	Attribute	M/O	Value	Comments
Start	StartOrder	M	Numeric	Line-up
	SortOrder	M	Numeric	
Composition /Athlete	Bib	O	String	Bib number

The following table describes in more detail the UnitInfo element in the case of Cycling Mountain Bike.

Element: UnitInfo			
Type	**Code**	**Value**	**Description**
UI_ST	CM_ENTRIES	Number	For @Type: Send proposed type
			For @Code: Send proposed code
			For @Value: Send the number of entries.
	CM_NOCS	Number	For @Type: Send proposed type
			For @Code: Send proposed code
			For @Value: Send the number of CGAs

For the table above, we have the following additional/summary information:

Type /Code	Description	Expected
UI_ST /CM_ENTRIES UI_ST /CM_NOCS	The numbers of entries and CGAs for the statistics	When was available

The following table describes in more detail the EventUnitEntry element in the case of Cycling Mountain Bike.

Element: Competitor /Start /Competitor /Composition /Athlete /EventUnitEntry			
Type	**Code**	**Value**	**Description**
EUE_CM	CM_IRM	CC @IRM	For @Type: Send proposed type
			For @Code: Send proposed code
			For @Value: Indicator as supplied by OVR for DNS or other possible results before the race.

For the table above, we have the following additional/summary information:

Type /Code	Description	Expected
EUE_CM /CM_IRM	Invalid result mark supplied by OVR before the race.	As soon as this information is available

4.1.2.6 Message sort

Please, follow the general definition.

4.1.3 Event Unit Results

4.1.3.1 Description

This message is the Event Unit Results message as described in the ODF General Messages Interface Document.

4.1.3.2 Header Values

The DocumentCode attribute in the ODF header will be sent according to the ODF Common Codes document (header values sheet).

4.1.3.3 Trigger and Frequency

Please, follow the general definition, taking also into account the following

- For intermediate results:
  - After last competitor of each lap
- For partials results:
  - After 10 competitors at the finish line
- Official results:
  - After the results for race are approved

4.1.3.4 Message Structure

The optional elements defined for this message in the ODF General Messages Interface Document that should be included in the case of Cycling Mountain Bike are:

- UnitDateTime
- UnitInfo
- Competitor /Composition /Athlete /ExtendedResults /ExtendedResult

4.1.3.5 Message Values

The following table lists the Event Unit Results optional and/or extended attributes (defined in the ODF General Messages Interface Document), as well as the attributes that have an extended definition.

Element	Attribute	M/O	Value	Comments
Result	Rank	O	String	Rank of the competitor in the corresponding event unit.
	RankEqual	O	Y	Y in the case of equalled rank
	ResultType	O	CC @ResultType	Result type (see codes section)
	IRM	O	CC @IRM	IRM for the particular event unit. Send just in the case @ResultType is IRM, or both time and IRM (see codes section)
	Result	O	HH:MM:SS 99:90:00	Result for the particular event unit.
	SortOrder	M	Numeric	This attribute is a sequential number with the order of the results for the particular event unit

Send UnitDateTime including also the @EndDate attribute

The following table describes in more detail the UnitInfo element in the case of Cycling Mountain Bike.

Element: UnitInfo					
Type	**Code**	**Extension Code**	**Pos**	**Value**	**Description**
UI_RESULTS	CM_AFTER		N(1) 0	String	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Send 1 for partial result Send 2 for race analysis
					For @Value: For partial result (@pos=1): x of Y Riders Or during the race (@pos=2): x riders have completed Lap y (z Km)
	CM_FASTEST		N(2) 90	S(20) with no leading zeroes	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Send the lap's number in which the competitor had the best time.
					For @Value: Send the competitor most faster
UI_ST	CM_FINISHED			Number	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Send the lap number who have completed that riders
					For @Value: Send number of riders who finish the race.
	CM_*y* Where y=CC@IRM			Number	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything.
					For @Value: Send number of riders who have IRM.
UI_LEADER	CM_CURRENT			N(2) 90	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything
					For @Value: Send the intermediate point was the current leader has most recently passed
		CM_ID		S(20) with no leading zeroes	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything
					For @Value: Send the Current Leader ID from the intermediate point
	CM_INTERMEDIATE		N(2) 90	HH:MM:SS 99:90:00	For @Type: Send proposed type

Element: UnitInfo					
Type	**Code**	**Extension Code**	**Pos**	**Value**	**Description**
					For @Code: Send proposed code
					For @Pos: The number that identifies the intermediate result point, from 1 to the total number (n) of intermediate result points. Where n is when finish the race. According to the @pos of the EC_RACE /CM_INTERMEDIATE code at the DT_CONFIG message
					For @Value: Leader Time up to that point
		CM_AVGSPEED		N(3).N(3) 990.000	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything
					For @Value: ~~Leader Average Speed up to that point~~ Average Speeds (from start) Km/h, The Speed from the start to an intermediate
		CM_LAVGSPEED		N(3).N(3) 990.000	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything
					For @Value: Average Speed (last lap) km/h, the lap speed from leader (start of lap) to leader (end of lap)
	CM_LAP		N(2) 90	HH:MM:SS 99:90:00	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: The number that identifies the lap, from 1 to the total number (n) of laps. ~~According to the @pos of the EC_RACE /CM_INTERMEDIATE code at the DT_CONFIG message~~ According to the @pos of the EC_RACE /CM_LAP code at the DT_CONFIG message
					For @Value: Time for that lap.
		CM_AVGSPEED		N(3).N(3) 990.000	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything
					For @Value:

Element: UnitInfo					
Type	**Code**	**Extension Code**	**Pos**	**Value**	**Description**
					Average Speed (from start) in that lap, the lap speed for the leader at the end of the lap

For the table above, we have the following additional/summary information:

Type /Code	Description	Expected
CM_AFTER	For partial result: "x of Y Riders " The number of riders who just passed in that moment and the total of riders. Or During the race: "x riders have completed Lap y (z Km)" Race distance completed.	When was available
CM_FASTEST	To know the faster competitor and in which lap happened	Send only at the end of the race
UI_ST /CM_FINISHED UI_ST /CM_y Where y = CC @IRM	Number of riders who have finish in that moment and the number of riders who have IRM.	When was available
UI_LEADER /CM_CURRENT	Send the intermediate point where the current leader has most recently passed.	when it is available
CM_CURRENT/CM_ID	Send the ID of the current Leader	
UI_LEADER /CM_INTERMEDIATE	Are points in the race, from 1 to n (finish). The @pos attribute should be according to the @pos of the EC_RACE /CM_INTERMEDIATE code at the DT_CONFIG message	when it is available
CM_INTERMEDIATE/ CM_AVGSPEED	"Average Speeds (from start) Km/h", Average speed from the time competitor X crosses the first intermediate (competitor X may be in the lead or not) until the time competitor X (now the leader) crosses the last intermediate point.	Only after each Lap
CM_INTERMEDIATE/ CM_LAVGSPEED	"Average Speeds (last lap) Km/h",Average speed from the time the leader at the first intermediate of a lap crosses this point until the time the leader crosses the last intermediate point of the lap.	Only after each Lap
UI_LEADER /CM_LAP	Are sections between intermediate points The @pos attribute should be according to the @pos of the EC_RACE /CM_LAP code at the DT_CONFIG message The @Value attribute is the Lap time.	when it is available
CM_LAP/ CM_AVGSPEED	Average speed from the start of the race until the time the leader crosses the last intermediate point of a lap.	

The following table describes in more detail the Competitor /Composition /Athlete /ExtendedResults /ExtendedResult element.

Element: Competitor /Composition /Athlete /ExtendedResults /ExtendedResult					
Type	**Code**	**Extension Code**	**Pos**	**Value**	**Description**
ER_IRM	CM_LAP			N(2) 90	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos:

Element: Competitor /Composition /Athlete /ExtendedResults /ExtendedResult					
Type	**Code**	**Extension Code**	**Pos**	**Value**	**Description**
					Do not send anything
					For @Value: If the @IRM=LAP Send the laps remaining for finish the race. If the @IRM=DNF Send the lap when the competitor left the race.
ER_RESULTS	CM_CURRENT			N(2) 90	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything
					For @Value: Intermediate point was the athlete has most recently passed If the competitor has an IRM: 1. In case the DNS or the athlete has an IRM before he crosses the first intermediate point: send 1. 2. In other cases, send the Intermediate point that he has crossed most recently plus 1.
	CM_SPRINTOFF			S(1)	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything
					For @Value: Send “Y” for the indication of the sprint-off winner
	CM_INTERMEDIATE		N(2) 90	HH:MM:SS 99:90:00	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Intermediate point where the competition has taken place (1,2..)
					For @Value: Cumulative time after the intermediate point (@pos)
		CM_RANK		String	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything
					For @Value: Cumulative Athlete’s rank after the intermediate point
		CM_ERANK		Y	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything
					For @Value: Y in the case of equalled rank
		CM_DIFF		+H:MM:SS (+9:90:00)	For @Type: Send proposed type

Element: Competitor /Composition /Athlete /ExtendedResults /ExtendedResult					
Type	**Code**	**Extension Code**	**Pos**	**Value**	**Description**
				Or 0:00 for the leader	For @Code: Send proposed code
					For @Pos: Do not send anything
					For @Value: The difference time between that competitor and the leader until that intermediate point.
		CM_AVRSPEED		N(3).N(3) 990.000	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything
					For @Value: Average Speed up to that point
		CM_IDX		N(3) 990	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything
					For @Value: Index based on whole list (with the ones who have not passed yet are indexed as well – after the ones who have finished, but before the IRMs. Sorted by the intermediate passed most recently and by order there (if none, then by start order)). Overlapped (LAP) riders must be dropped to the bottom but above the DNF/DSQ/DNS riders. For tied athletes, the rider with the lowest bib number is listed first.
		CM_PASSIDX		N(3) 990	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything
					For @Value: Index based on only the ones who have finished or have an IRM
		CM_LASTFINISH		S(1)	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything
					For @Value: Send “Y” for the competitor who has just finished the split or the race.
		CM_OFFICIAL		S(1)	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything
					For @Value: Send Y when the time becomes official Send N when the time is unofficial.
	CM_LAP		N(2) 90	HH:MM:SS 99:90:00	For @Type: Send proposed type

Element: Competitor /Composition /Athlete /ExtendedResults /ExtendedResult					
Type	**Code**	**Extension Code**	**Pos**	**Value**	**Description**
					For @Code: Send proposed code
					For @Pos: The number that identifies the lap, from 1 to the total number of laps.
					For @Value: Time for that lap.
		CM_RANK		String	For @Type: Send proposed type
					For @Pos: Do not send anything
					For @Code: Send proposed code
					For @Value: Athlete's rank for that Lap.
		CM_ERANK		Y	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything
					For @Value: Y in the case of equalled rank
		CM_DIFF		+H:MM:SS (+9:90:00) Or 0:00 for the leader	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything
					For @Value: Time difference up to that point. Send only for athletes in the split of the leader.
		CM_AVGSPEED		N(3).N(3) 990.000	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything
					For @Value: Average Speed up to that point
		CM_IDX		N(3) 990	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything
					For @Value: Index based on whole list (with the ones who have not passed yet are indexed as well – after the ones who have finished, but before the IRMs. Sorted by the intermediate passed most recently and by order there (if none, then by start order)). Overlapped (LAP) riders must be dropped to the bottom but above the DNF/DSQ/DNS riders. For tied athletes, the rider with the lowest bib number is listed first.
		CM_PASSIDX		N(3) 990	For @Type: Send proposed type
					For @Code: Send proposed code

Element: Competitor /Composition /Athlete /ExtendedResults /ExtendedResult					
Type	**Code**	**Extension Code**	**Pos**	**Value**	**Description**
					For @Pos: Do not send anything
					For @Value: Index based on only the ones who have finished or have an IRM
		CM_LASTFINISH		S(1)	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything
					For @Value: Send “Y” for the competitor who has just finished the split
	CM_SLOOP		N(1) 0	HH:MM:SS 99:90:00	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Send the number that identifies that start loop.
					For @Value: Send the time for the start loop, the best time should be bolded.
		CM_RANK		String	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything
					For @Value: Athlete’s start loop rank for that start Loop.
		CM_ERANK		Y	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything
					For @Value: Y in the case of equalled rank
	CM_PHOTO			S(1)	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything
					For @Value: To know if the competitor’s final result was decided by photo. Send Y for Evaluated Status Send P for Pending Status.

For the table above, we have the following additional/summary information:

Type /Code	Description	Expected
ER_IRM /CM_LAP	Laps remaining or laps when the competitor left the race. Example: If the rider abandoned in lap 0 @IRM=”DNF” CM_IRM@Value=0	Only If the @IRM=LAP or @IRM=DNF Only if the results are approved.

	If the rider abandoned in lap 3 @IRM=”DNF” CM_IRM@Value=3 If the rider lapped with 4 laps remaining @IRM=”LAP” CM_IRM@Value=4	
ER_RESULTS /CM_CURRENT	Send for the athlete who has just finished the intermediate point or the race. If the competitor has an IRM: 1. In case the DNS or the athlete has an IRM before he crosses the first intermediate point: send 1. 2. In other cases, send the Intermediate point that he has crossed most recently plus 1.	Always
ER_RESULTS /CM_SPRINTOFF	Indication of the sprint-off winner	If it is necessary
ER_RESULTS /CM_INTERMEDIATE	It’s the intermediate point where the competitor has taken place (Time, Rank, Avr. Speed, Difference time).	when it is available
CM_INTERMEDIATE /CM_OFFICIAL	Use for show if the results are official.	Always
ER_RESULTS /CM_LAP	Section between the intermediate points. The section 1 not included the Start Loop information	when it is available
CM_RANK CM_ERANK	Cumulative Athlete’s rank after that point	when it is available
CM_DIFF	Intermediate Time difference.	when it is available
CM_AVRSPEED	Cumulative Avr. Speed	when it is available
CM_IDX	Index based on whole list (with the ones who have not passed yet are indexed as well – after the ones who have finished, but before the IRMs. Sorted by the intermediate passed most recently and by order there (if none, then by start order)) of each intermediate point. This element is like a historical of @SortOrder. **Example:** If the leader (ATH1) is in the intermediate point 3 and the Athlete AT2 just to pass for that point so the values for these athletes are **AT1** @SortOrder 1 INTERMEDIATE 3 / _IDX 1 **AT2** @SortOrder 2 _INTERMEDIATE 3 / _IDX 2 Then If the Athlete AT2 is the first to pass for the intermediate point 4, he will be the new leader in that point so the values for these athletes are: **AT2** @SortOrder 1 _INTERMEDIATE 3 / _IDX 2 _INTERMEDIATE 4 / _IDX 1 **AT1** @SortOrder 2 _INTERMEDIATE 3 / _IDX 1 _INTERMEDIATE 4 / _IDX 2	Always
CM_PASSIDX	Index based on only the ones who have finished or have an IRM. The difference between _IDX and _PASSIDX is	If applies.

	the first one will be the order of all the riders who have passed an specific intermediate point or are behind this or have an IRM, while the second one has only the riders who have passed an specific intermediate point or have IRM Example: If we have 10 riders, and 5 have passed through the point 3, both as _IDX @SortOrder contain the order of 10 riders, while the _PASSIDX only be sent to the 5 riders who just passed.	
CM_LASTFINISH	Use this for the competitor who has just finished the intermediate point/Lap or the race.	If applies.
ER_RESULTS /CM_SLOOP	Section between the intermediate points for the start loop. (Time, Rank for start loop)	Only included if start loops are actually used.
CM_SLOOP /CM_RANK CM_SLOOP /CM_ERANK	Athlete's start loop rank for that start Loop.	when it is available
ER_RESULTS /CM_PHOTO	It is an attribute for know if it is necessary made a photo for this competitor.	At the end of the race. Only send for competitor who needs that.

4.1.3.6 Message sort

Please, follow the general definition.

4.1.4 Official Communication

4.1.4.1 Description

This message is the Official Communication message as described in the ODF General Messages Interface Document.

4.1.4.2 Header Values

The DocumentCode attribute in the ODF header will be sent according to the ODF Common Codes document (header values sheet).

In case of Race Incidents send @DocumentSubcode with “RINCIDENT” and @DocumentSubcode always with “1”.

4.1.4.3 Trigger and Frequency

- After each incident

4.1.4.4 Message Structure

The optional elements defined for this message in the ODF General Messages Interface Document that should be included in the case of Cycling Mountain Bike are:

- JuryDecision (following the general rules for this element)
- RIncidents (following the general rules for this element and Send only for the race incidents)

4.1.4.5 Message Values

Please, follow the general definition.

4.1.4.6 Message sort

Please, follow the general definition.

4.1.5 Discipline Configuration

4.1.5.1 Description

This message is the Discipline Configuration message as described in the ODF General Messages Interface Document.

4.1.5.2 Header Values

Please, follow the general definition.

4.1.5.3 Trigger and Frequency

Please, follow the general definition.

4.1.5.4 Message Structure

The optional elements defined for this message in the ODF General Messages Interface Document that should be included in the case of Cycling Mountain Bike are:

- ExtendedConfigItem

4.1.5.5 Message Values

Send the attributes and codes according to the tables described in this section.

The following table lists the Discipline configuration optional attributes (defined in the ODF General Messages Interface Document) that are used in the case of Cycling Mountain Bike, as well as the attributes that have an extended definition.

Element	Attribute	M/O	Value	Comments
Config	Gender	O	CC @Gender	
	Event	O	CC @Event	

The following table describes in more detail the ExtendedConfig element.

Element: ExtendedConfig					
Type	**Code**	**ExtendedConfig Item Code**	**Pos**	**Value**	**Description**
EC_ENTRY	CM_UCIDATE (By discipline)			YYYYMMDD	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything.
					For @Value: Send the date for UCI Ranking
EC_RACE (By event)	CM_RDISTANCE			Number	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything.
					For @Value: Send the total distance for the race in km.
		CM_T_LAPS		N(2) 90	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything
					For @Value: Total Lap's numbers in the race

Element: ExtendedConfig					
Type	**Code**	**ExtendedConfig Item Code**	**Pos**	**Value**	**Description**
		CM_T_SLOOP		N(1) 0	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything
					For @Value: Total Start Loop in the race Send if exist a start Loop
		CM_T_INTER		N(2) 90	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything
					For @Value: Total intermediate points in the race(not including the intermediate 0)
		CM_FLOOP		N(1) 0	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything
					For @Value: Total Finish Loops in the race Send if a finish loop exists
	CM_INTERMEDIATE		N(2) 90		For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Send the number that identifies the intermediate point, from 1 to n. Where n is when finish the race.
					For @Value: Do not send anything.
		CM_DISTANCE		N(3).N(1) 999.0	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything.
					For @Value: Send distance in km
		CM_IS_LAST		S(1)	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything.
					For @Value: Send “F”. Only send for the last intermediate point (when finish the race).
		CM_TYPE		CC @IntPtType	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything.

Element: ExtendedConfig					
Type	**Code**	**ExtendedConfig Item Code**	**Pos**	**Value**	**Description**
					For @Value: Send an indication of whether the timing point is the Start Loop (SL), a Half Lap (HL), a Lap (LAP), or the Finish Loop (FL) (see codes section)
		CM_TYPE_DISTANCE		N(2).N(1) 90.0	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything.
					For @Value: Send an indication of whether the timing point is a 'Start Loop', an 'Intermedate Timing Point', or a full 'Lap'. Where the CM_TYPE is SL (Start Loop) or FL (Finish Loop) then do not send the CM_TYPE_DISTANCE attribute, for the 'Half Lap' the value is "*x*.5" where *x* is the number of completed laps, and for the 'Lap' the value is "*x*.0" where *x* is the number of completed laps.
	CM_LAP		N(2) 90		For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: The number that identifies the lap, from 1 to the total number of lap.
					For @Value: Do not send anything.
		CM_DISTANCE		N(3).N(2) 999.00	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything.
					For @Value: Lap's distance in km
		CM_START		N(2) 90	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything
					For @Value: Send the intermediate point when start the lap.
		CM_FINISH		N(2) 90 Or S(1)	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything
					For @Value: Send the intermediate point when finish the lap. For lap n, send "F", where n is the point in the race. For the last lap send "F"
	CM_SLOOP		N(2)		For @Type:

Element: ExtendedConfig					
Type	**Code**	**ExtendedConfig Item Code**	**Pos**	**Value**	**Description**
			90		Send proposed type
					For @Code: Send proposed code
					For @Pos: Send the number that identifies the start loop.
					For @Value: Do not send anything.
		CM_DISTANCE		N(3).N(2) 999.00	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything.
					For @Value: Send distance in km.
		CM_START		N(2) 90	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything
					For @Value: Send the intermediate point when start the start loop.
		CM_FINISH		N(2) 90 Or S(1)	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything
					For @Value: Send the intermediate point when finish the start loop. For start loop n, send “F”, where n is the point in the race.

For the table above, we have the following additional/summary information

Type /Code	Description	Expected
E_ENTRY /CM_UCIDATE	Send the date for UCI Ranking	As soon as the venue results has this information
EC_RACE /CM_RDISTANCE	Race description(total race distance, total number of Laps, start loop, finish loop and intermediate points)	Before the start of the race
EC_RACE /CM_INTERMEDIATE	Are points in the race where taken results (Example: after start loop, after each laps, after the finish loop, at the end of the race, ..), from 0 to n. Where 0 is when start the race (at this point does not send results only to have a reference of where to start) and n is when finish the race. Example: If the race has: 1 Start Loop + 2 Lap + 1 Finish loop and we will need to know the times in all that points, the description should be like bellow: Race description: _T_SLOOP=1 _T_LAPS=2 _FLOOP=1 _T_INTER=4, because we will take times in all the previous	when it is available

	points _SLOOP =1, _START = 0, _FINISH = 1 _LAP = 1, _START = 1, _FINISH = 2 _LAP = 2, _START = 2, _FINISH = 3 _LAP = 3, _START = 3, _FINISH = 4 (This is the Finish Loop) _INTERMEDIATE = 1,2,3 0 when start the race, but this will not be sent. 1 after the start loop 2,3 after each lap 4 after the finish loop (at the end of the race), in this case send a "_IS_LAST" with "F".	
EC_RACE /CM_INTERMEDIATE/CM_TYPE EC_RACE /CM_INTERMEDIATE /CM_TYPE_DISTANCE	Define whether the timing point is the Start Loop (SL), a Half Lap (HL), a Lap (LAP), or the Finish Loop (FL) Example: If we have 3 intermediate points, where the first was made in the Start loop, the second after half lap and the third after 1 lap: _INTERMEDIATE = 1 _TYPE=SL _INTERMEDIATE = 2 _TYPE=HL _TYPE_DISTANCE=0.5 _INTERMEDIATE = 3 _TYPE=LAP _TYPE_DISTANCE=1.0 CM_TYPE_DISTANCE - Where the CM_TYPE is SL (Start Loop) or FL (Finish Loop) then do not send the CM_TYPE_DISTANCE attribute, for the 'Half Lap' the value is "x.5" where x is the number of completed laps, and for the 'Lap' the value is "x.0" where x is the number of completed laps.	If apply
EC_RACE /CM_LAP CM_LAP/ CM_DISTANCE	They are the sections for the laps and the finish loop in the race. Example: If the race has: 1 Start Loop + 2 Lap + 1 Finish loop We will have 3 elements _LAP _LAP@pos=1 _LAP@pos=2 _LAP@pos=3 If the race has: Start Loop: 1 x 0.29km Laps: 7 x 5.00km Finish Loop: 1 x 0.31km Total Race Distance: 35.6Km We will have 8 elements _LAP _LAP@pos=1 _DISTANCE@Value=5.00 _LAP@pos=2 _DISTANCE@Value=5.00 _LAP@pos=3 _DISTANCE@Value=5.00 … _LAP@pos=7 _DISTANCE@Value=5.00 _LAP@pos=8 _DISTANCE@Value=0.31 If the race has: Start Loop: 1 x 0.60km Laps: 7 x 5.00km Total Race Distance: 35.6Km We will have 7 elements _LAP	when it is available

	_LAP@pos=1 _DISTANCE@Value=5.00 _LAP@pos=2 _DISTANCE@Value=5.00 _LAP@pos=3 _DISTANCE@Value=5.00 … _LAP@pos=7 _DISTANCE@Value=5.00	
CM_LAP/CM_START CM_LAP/CM_FINISH	To know in which intermediate points start or finish that section. Example: If the race has: 1 start loop, 2 laps and one finish loop The _LAP@pos=1 starts in the intermediate point “1” and finish in the “2”. The _LAP@pos=2 starts in the intermediate point “2” and finish in the “3”. The _LAP@pos=3 starts in the intermediate point “3” and finish in the “4”.	when it is available
CM_SLOOP CM_SLOOP/ CM_DISTANCE	Are sections for the start loops in the race. Example: If the race has: 1 start loop, 2 laps and one finish loop _SLOOP@pos=1 If the race has: Start Loop: 1 x 0.29km Laps: 7 x 5.00km Finish Loop: 1 x 0.31km We will have 1 elements _SLOOP _SLOOP@pos=1 _DISTANCE@Value=0.29 If the race has: Start Loop: 1 x 0.60km Laps: 7 x 5.00km Total Race Distance: 35.6Km We will have 3 elements _SLOOP _SLOOP@pos=1 _DISTANCE@Value=0.60	when it is available
CM_SLOOP/CM_START CM_SLOOP/CM_FINISH	To know in which intermediate points start or finish that section. Example: If the race has: 1 start loop, 2 laps and one finish loop The _SLOOP@pos=1 start in the intermediate point 0 and finish in 1.	when it is available

4.1.5.6 Message sort

Please, follow the general definition.

4.1.6 Event Unit Weather Conditions

4.1.6.1 Description

This message is the Event Unit Weather Conditions message as described in the ODF General Messages Interface Document.

4.1.6.2 Header Values

The DocumentCode attribute in the ODF header will be sent according to the ODF Common Codes document (header values sheet).

4.1.6.3 Trigger and Frequency

Please, follow the general definition.

4.1.6.4 Message Structure

The optional elements defined for this message in the ODF General Messages Interface Document that should be included in the case of Cycling Mountain Bike are:

- Competition /Weather /Conditions /Condition (following the general rules for this element)

4.1.6.5 Message Values

The following table lists the Event Unit Weather Conditions optional and/or extended attributes (defined in the ODF General Messages Interface Document), as well as the attributes that have an extended definition

Element	Attribute	M/O	Value	Comments
Conditions	Code	M	GL	GL for generically, because this information will only be measured once.
	Humidity	M	N(3) 990	Humidity in %
Competition /Weather /Conditions /Condition	Code	M	SKY	Weather condition type
	Value	M	CC @WeatherCondition	Codes that describe the weather
Competition /Weather /Conditions /Temperature	Code	M	AIR	Air
	Unit	M	CC @TemperatureUnit	Metric system unit for temperature
	Value	M	N(2) 90	Temperature of the @Code

4.1.6.6 Message sort

Please, follow the general definition.

5 Real time

The following chapter describes the ODF-RT part of Cycling Mountain Bike.

5.1 Real Time Applicable Messages

The next table is a full list of all ODF-RT messages and describes the list of messages used in Cycling Mountain Bike the same way as it is done in the table of chapter 4.

Message Type	Message name	Message used in this sport	Message extended in this document
DT_RT_RESULT	RT Event Unit Results	X	X
DT_RT_CUMULATIVE_RESULT	RT Cumulative Results		
DT_RT_CLOCK	RT Clock		
DT_RT_GM	RT Discipline/Venue good morning	X	
DT_RT_GN	RT Discipline/venue good night	X	
DT_RT_KA	RT Discipline/venue keep alive	X	

5.1.1 RT Event Unit Results

5.1.1.1 Description

This message is the RT Event Unit Results message as described in the ODF General Messages Interface Document.

5.1.1.2 Header Values

The ODF header will be sent according to the ODF Common Codes document.

5.1.1.3 Trigger and Frequency

The following is the trigger for this message in ODF-RT:

- ResultStatus=”LIVE_UPDATE”
  - T1: Trigger when status changes.
  - T2: Trigger when an athlete passes by the intermediate point.
  - T3: Trigger when leader reach an intermediate point.
  - T4: Trigger during the race if the Status is required.
  - T5: Trigger throughout the race.
  - T6: Trigger after each competitor finishes.
- ResultStatus=”LIVE_FULL”
  - This value should be suggested after further testing and sent in the DT_RT_GM message after further testing
  - Suggested to test frequency values around the average time used by the athlete to complete their participation in the course.
- for the other ResultStatus, please, follow the general definition.

5.1.1.4 Message Structure

The optional elements defined for this message in the ODF General Messages Interface Document that should be included in the case of Cycling Mountain Bike are:

- UnitInfo
- Competitor /Composition /Athlete /ExtendedResults /ExtendedResult

Please, follow the general considerations for all the different type of messages.

In the next section (message values), there is a more detailed definition.

5.1.1.5 Message Values

The following table describes in more detail the Result element.

Element	Attribute	M/O	Value	Comments	LIVE_UPDATE RT trigger expected
Result	Rank	O	String	Rank of the competitor in the corresponding event unit.	T5(Only for those athletes who are at the same split as the leader), T1(If applies)
	RankEqual	O	Y or N	Y in the case of equalled rank. N in the case of tie break.	
	ResultType	O	CC @ResultType	Result type (see codes section)	T2, T4, T5, T1, T3

Element	Attribute	M/O	Value	Comments	LIVE_UPDATE RT trigger expected
	IRM	O	CC @IRM	IRM for the particular event unit Send just in the case @ResultType is IRM, or both time and IRM (see codes section)	T1, T4
	Result	O	HH:MM:SS 99:90:00	Result for the particular event unit.	T6, T1
	SortOrder	M	Numeric	This attribute is a sequential number with the order of the results for the particular event unit	T2, T4, T5,T6

The following table describes in more detail the UnitInfo element in the case of Cycling Mountain Bike.

Element: UnitInfo					
Type	**Code**	**Extension Code**	**Pos**	**Value**	**Description**
UI_LEADER	CM_CURRENT			N(2) 90	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything
					For @Value: Send the intermediate point was the current leader has most recently passed.
		CM_ID		S(20) with no leading zeroes	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything
					For @Value: Send the Current Leader ID from the intermediate point
	CM_INTERMEDIATE		N(2) 90	HH:MM:SS 99:90:00	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: The number that identifies the intermediate result point, from 1 to the total number (n) of intermediate result points. Where n is when finish the race. According to the @pos of the EC_RACE /CM_INTERMEDIATE code at the DT_CONFIG message
					For @Value: Leader Time up to that point
		CM_AVGSPEED		N(3).N(3) 990.000	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything
					For @Value: Average Speeds (from start) Km/h, The Speed from the start to an intermediate
		CM_LAVGSPEED		N(3).N(3) 990.000	For @Type: Send proposed type
					For @Code: Send proposed code

Element: UnitInfo					
Type	**Code**	**Extension Code**	**Pos**	**Value**	**Description**
					For @Pos: Do not send anything
					For @Value: Average Speed (last lap) km/h, the lap speed from leader (start of lap) to leader (end of lap)
	CM_LAP		N(2) 90	HH:MM:SS 99:90:00	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: The number that identifies the lap, from 1 to the total number (n) of laps. According to the @pos of the EC_RACE /CM_LAP code at the DT_CONFIG message
					For @Value: Time for that lap.
		CM_AVGSPEED		N(3).N(3) 990.000	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything
					For @Value: Average Speed (from start) in that lap, the lap speed for the leader at the end of the lap

For the table above, we have the following additional/summary information:

Type /Code	Description	Expected
UI_LEADER /CM_CURRENT	Send the intermediate point where the current leader has most recently passed.	when it is available
CM_CURRENT/CM_ID	Send the ID of the current Leader	
UI_LEADER /CM_INTERMEDIATE	Are points in the race, from 1 to n (finish). The @pos attribute should be according to the @pos of the EC_RACE /CM_INTERMEDIATE code at the DT_CONFIG message	T3
CM_INTERMEDIATE/ CM_AVGSPEED	“Average Speeds (from start) Km/h”, Average speed from the time competitor X crosses the first intermediate (competitor X may be in the lead or not) until the time competitor X (now the leader) crosses the last intermediate point.	T3
CM_INTERMEDIATE/ CM_LAVGSPEED	“Average Speeds (last lap) Km/h”, Average speed from the time the leader at the first intermediate of a lap crosses this point until the time the leader crosses the last intermediate point of the lap.	T3, but only for those intermediate points (CM_INTERMEDIATE@pos) that match with a Lap.
UI_LEADER /CM_LAP	Are sections between intermediate points. The @pos attribute should be according to the @pos of the EC_RACE /CM_LAP code at the DT_CONFIG message The @Value attribute is the Lap time.	T3, but only for those intermediate points that match with a Lap.
CM_LAP/ CM_AVGSPEED	Average speed from the start of the race until the time the leader crosses the last intermediate point of a lap.	T3, but only for those intermediate points that match with a Lap.

The following table describes in more detail the Competitor /Composition /Athlete /ExtendedResults /ExtendedResult element.

Element: Competitor /Composition /Athlete /ExtendedResults /ExtendedResult					
Type	**Code**	**Extension Code**	**Pos**	**Value**	**Description**
ER_RESULTS	CM_CURRENT			N(2) 90	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything
					For @Value: Intermediate point was the athlete has most recently passed. If the competitor has an IRM: 1. In case the DNS or the athlete has an IRM before he crosses the first intermediate point: send 1. 2. In other cases, send the Intermediate point that he has crossed most recently plus 1.
	CM_INTERMEDIATE CM_LAP		N(2) 90	HH:MM:SS 99:90:00	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: The number that identifies lap/intermediate, from 1 to the total numbers for that(For intermediate It is when finish the race)
					For @Value: Time after the Lap/intermediate
		CM_RANK		String	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything
					For @Value: Rank up to that point
		CM_ERANK		Y or N	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything
					For @Value: Y in the case of equalled rank. N in the case of tie break.
		CM_DIFF		+H:MM:SS (+9:90:00) Or 0:00 for the leader	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything
					For @Value: Time difference up to that point. Send only for athletes in the split of the leader. Display leader as 0:00.
		CM_AVGSPEED		N(3).N(3) 990.000	For @Type: Send proposed type
					For @Code:

Element: Competitor /Composition /Athlete /ExtendedResults /ExtendedResult					
Type	**Code**	**Extension Code**	**Pos**	**Value**	**Description**
					Send proposed code
					For @Pos: Do not send anything
					For @Value: Average Speed up to that point
		CM_IDX		N(3) 990	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything
					For @Value: Index based on whole list (with the ones who have not passed yet are indexed as well – after the ones who have finished, but before the IRMs. Sorted by the intermediate passed most recently and by order there (if none, then by start order)). Overlapped (LAP) riders must be dropped to the bottom but above the DNF/DSQ/DNS riders. For tied athletes, the rider with the lowest bib number is listed first.
		CM_PASSIDX		N(3) 990	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything
					For @Value: Index based on only the ones who have finished or have an IRM
		CM_LASTFINISH		S(1)	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything
					For @Value: Send “Y” for the competitor who has just finished the split or the race. Send N if it is not more.
		CM_OFFICIAL		S(1)	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything
					For @Value: Send Y when the time becomes official Send N when the time is unofficial. Only for Intermediate points.
	CM_PHOTO			S(1)	For @Type: Send proposed type
					For @Code: Send proposed code
					For @Pos: Do not send anything
					For @Value: To know if the competitor’s final result was decided by photo. Send Y for Evaluated Status Send P for Pending Status.

Element: Competitor /Composition /Athlete /ExtendedResults /ExtendedResult					
Type	**Code**	**Extension Code**	**Pos**	**Value**	**Description**
					Send N if it has not more

For the table above, we have the following additional/summary information:

Type /Code	Description	Expected
ER_RESULTS /CM_CURRENT	Send for the athlete who has just finished the intermediate point or the race. If the competitor has an IRM: 1. In case the DNS or the athlete has an IRM before he crosses the first intermediate point: send 1. 2. In other cases, send the Intermediate point that he has crossed most recently plus 1.	Always
ER_RESULTS /CM_INTERMEDIATE	Are points in the race where we have the competitor's results (time, rank, difference, avr, speed).	when it is available
CM_INTERMEDIATE /CM_OFFICIAL	Use for show if the results are official	Always
ER_RESULTS /CM_LAP	Section between two intermediate points.	when it is available
CM_RANK CM_ERANK	Rank up to that point	when it is available
CM_DIFF	Time difference up to that point	
CM_AVRSPEED	Average Speed up to that point	
CM_IDX	Index based on whole list (with the ones who have not passed yet are indexed as well – after the ones who have finished, but before the IRMs. Sorted by the intermediate passed most recently and by order there (if none, then by start order)) of each intermediate point. This element is like a historical of @SortOrder. **Example:** If the leader (ATH1) is in the intermediate point 3 and the Athlete AT2 just to pass for that point so the values for these athletes are **AT1** @SortOrder 1 _INTERMEDIATE 3 / _IDX 1 **AT2** @SortOrder 2 _INTERMEDIATE 3 / _IDX 2 Then If the Athlete AT2 is the first to pass for the intermediate point 4, he will be the new leader in that point so the values for these athletes are: **AT2** @SortOrder 1 _INTERMEDIATE 3 / _IDX 2 _INTERMEDIATE 4 / _IDX 1 **AT1** @SortOrder 2 _INTERMEDIATE 3 / _IDX 1 _INTERMEDIATE 4 / _IDX 2	Always
CM_PASSIDX	Index based on only the ones who have finished or have an IRM.	If applies.

	The difference between _IDX and _PASSIDX is the first one will be the order of all the riders who have passed an specific intermediate point or are behind this or have an IRM, while the second one has only the riders who have passed an specific intermediate point or have IRM Example: If we have 10 riders, and 5 have passed through the point 3, both as _IDX @SortOrder contain the order of 10 riders, while the _PASSIDX only be sent to the 5 riders who just passed.	
CM_LASTFINISH	Use this for the competitor who has just finished the intermediate point/Lap or the race.	If applies.
ER_RESULTS /CM_PHOTO	It is an attribute for know if it is necessary made a photo for this competitor.	At the end of the race. Only send for competitor who needs that.

5.1.1.6 Message sort

Please, follow the general definition.

6 PDF feed

Please refer to the same section of the ODF General Messages Interface Document.

DOCUMENT CONTROL

Version history

Version	Date	Comments
R1 v1.0	15 March 2013	First version SFR
R1 v1.1	12 April 2013	SFA
R1 v1.2	19 April 2013	APP
R1 v1.3	1 August 2013	APP
R1 v1.4	22 May 2014	APP
R1 v1.5	16 June 2014	External delivery

File reference: ODF/INT123 R1 v1.5 APP (CM)

Change Log

Version	Status	Changes on version
R1 v1.0	SFR	• First version
R1 v1.1	SFA	• Submitted for Approval.
R1 v1.2	APP	• Approved version
R1 v1.3	APP	• Defect 94963: Clarification for the "Lap Time" (UI_LEADER\ CM_LAP@Value), "Average Speeds (from start) Km/h" (CM_INTERMEDIATE/ CM_AVGSPEED), "Average Speeds (last lap) Km/h" (CM_INTERMEDIATE/ CM_LAVGSPEED). • Defect 98163: The EC_QUOTA from CT_CONFIG is not needed anymore • External Delivery
R1 v1.4	APP	• CR350: Remove the UCI Ranking (E_RANK) from the Participant message
R1 v1.5	APP	• External delivery

This page has been intentionally left blank