

Technical Annex: Calculation of Final Achievement Factors for 2006/07 and Provisional Achievement Factors for 2007/08 including SPSS pseudo code (to Reflect Changed Assumed Fee Rate for 2006/07 and 2007/08)

- 1 This document explains the calculation of 2006/07 and 2007/08 achievement factors for Further Education providers. Due to the change in fee assumption from 27.5 per cent to 32.5 per cent in 2006/07 and to 37.5 per cent in 2007/08 and the consequential change to programme funding, it is necessary to revise the methodology used to calculate achievement factors year on year, since the achievement factor is calculated from the programme funding. The change in fee assumption in 2006/07 now requires the LSC to estimate what the levels of programme funding would have been in a previous year if the fee assumption had been 32.5 per cent.
- 2 For 2006/07, achievement funding continues to be based on the ratio of achievement funding to programme funding from the most recent complete data for the institution. The 2006/07 achievement factor is calculated from the 2005/06 ILR F05.
- 3 To calculate indicative funding from the start of the 2007/08 teaching year, a provisional 2007/08 factor was derived using the 2005/06 ILR F05. The provisional factor will be updated with a final factor, used to calculate final funding, from final 2006/07 ILR F05 data returns.

Example

- 4 Table 1 below shows an example of the change in funding arising from the change in the fee element. This shows a learner with fee remission, studying and achieving 1 learning aim that attracts an unweighted base rate of £100 with programme weighting factor (PWF) = 1.3, disadvantage factor (DISF) = 1.1, area cost factor (ACF) = 1 and performance related factor (PRF) = 1. The achievement funding shown is as it would be calculated from the ILR achievement fields, rather than using the achievement factor.

Table 1: Example of Change in Funding Resulting from the fee element increase from 25% to 37.5%

	Weighted base rate	Programme funding	Assumed fee element / fee remission	Achievement element	Total funding
25% fee assumption	£143	£103.70	£25	£14.30	£143
27.5% fee assumption	£143	£101.20	£27.50	£14.30	£143
32.5% fee assumption	£143	£96.20	£32.50	£14.30	£143
37.5% fee assumption	£143	£91.20	£37.50	£14.30	£143
Calc	BASERATE * PWF * DISF * ACF * PRF.	WEIGHTED BASERATE – ASSUMED FEE ELEMENT – ACHIEVEMENT ELEMENT	FEE ASSUMPTION * UNWEIGHTED BASE RATE	10% * WEIGHTED BASE RATE	PROGRAMME FUNDING + ACHIEVEMENT FUNDING + FEE REMISSION

- 5 Calculation of each provider's final 2006/07 achievement factor is the summation of their ILR calculated achievement funding in 2005/06, divided by the summation of the revised 2005/06 programme funding, shown below.

$$\text{Achievement factor} = \frac{\text{ILR calculated achievement funding}}{\text{Programme funding adjusted to incorporate 32.5\% fee assumption}}$$

- 6 In the LIS for 2006/07, achievement funding in 2006/07 is calculated by multiplying the achievement factor by the 2006/07 programme funding for each aim, including the pseudo aim generated.
- 7 The calculation of the revised programme funding (with a fee assumption at 32.5 per cent for 2006/07 and at 37.5 per cent for 2007/08) is straightforward for the majority of learning aims:
- In 2006/07 the percentage change in fee element is calculated as $(32.5-27.5)/27.5$, which is 0.18181818 (recurring) and in 2007/08 this is $(37.5-27.5)/27.5$, which is 0.36363636 (recurring).
 - Where a franchise discount is applied, the revised franchised discount value is calculated by multiplying the old fee element by 0.06 for 2006/07 and 0.12 for 2007/08, then subtracting this value from the old franchise discount.
 - The programme funding is reduced by the difference between the current fee element and the revised fee element.
 - Where applicable, fee remission funding is increased by the difference between the current fee element and the revised fee element (this is only used when calculating the revised funding to be included in the additional enrichment record).
- 8 Complexity is increased for full-time learners aged 16-18 and eligible for enrichment funding. The total amount of programme and fee remission for (key skills + enrichment) does not change when the fee assumption changes. The funding for key skills changes (as in the formula above), but there is a balancing change in the opposite direction to the enrichment per learner. Although the total of (fee remission + programme funding) does not change, there will be less programme funding (and

more fee remission) so there is a complex calculation to calculate the revised programme funding.

Overview of calculation for enrichment learners

- a. Calculate revised programme funding for learning aims.
 - b. Calculate revised fee remission funding for learning aims.
 - c. Calculate total current amount of enrichment funding.
 - d. Total enrichment minus funding for keys skills equals total funding within additional enrichment record.
 - e. Fee remission for total enrichment minus fee remission for key skills equals fee remission within additional enrichment record.
 - f. Total funding within additional enrichment record minus fee remission within additional enrichment record equals programme funding within additional enrichment record.
- 9 Further details of the calculations can be found in the pseudo code which is located below.
- 10 The following exclusion applies to this calculation:
- a. where the funding per period for key skills is very close to the standard total for (key skills + enrichment), the revised programme funding may be a very small underestimate. However this situation is only likely to affect a very small number of learners and the effect on funding per learner is also very slight.

Pseudo code to calculate 2006/07 final achievement factors and 2007/08 provisional factors both based on F05 0506 ILR returns.
Amendments to pseudo code shown to calculate 2007/08 provisional Achievement factors.
Final 2006/07 Achievement factor calculation based on 2005/06 ILR F05.
Provisional 2007/08 Achievement factor calculation based on 2005/06 ILR F05.
Final 2007/08 factor will be calculated using 2006/07 ILR F05 once available.
*** 15 February 2007***.

Naming Convention

p1= Period 1
p2 = Period 2
p3 = Period 3
pn = Operation needs to be repeated for each period
KS = Key skills Learning aim
ORD = actual learning aim (excludes notional enrichment funding aims)
ENR = additional enrichment funding record
Fee_elem = Current fee element at 27.5% fee assumption
Corefund = Programme funding at 27.5% fee assumption
Feerem = Fee remission at 27.5% fee assumption
Franchdi = Franchised discount at 27.5% fee assumption
New_Fee_elem = Revised fee element at 32.5% fee assumption in 2006/07 calculation and 37.5% in 2007/08
New_Corefund = Revised Programme funding at 32.5% fee assumption
New_Feerem = Revised Fee remission at 32.5% fee assumption in 2006/07 calculation and 37.5% in 2007/08
New_Franchdi = Revised Franchised discount at 32.5% fee assumption

Parameters

Current fee element at 27.5% = !Current_perc_fee_elem = 0.275
New fee element at 32.5% (37.5% in 2007/08) = !new_perc_fee_elem = 0.325 in 2006/07 or 0.375 in 2007/08
**if the fee proportion changes from 27.5% to 32.5%, the franchised discount is $0.33*(32.5 - 27.5)/27.5 = 0.06$ **.
**if the fee proportion changes from 27.5% to 37.5%, the franchised discount is $0.33*(37.5 - 27.5)/27.5 = 0.12$ **.
Franchised Discount factor = !number3 = 0.06 in 2006/07 and 0.12 in 2007/08
Period enrichment funding in 2006/07 = period_enrichment_amount = 261.333333.

***Starting with Period / aim level file ***.

get file =\\ILR0506_F05_FUNDING_DATA_FULL.sav'.

Select year = 2005 funding only when calculating final 2006/07 factors with ILR 2005/06 F05.
Select year = 2005 funding only when calculating provisional 2007/08 factors with ILR 2005/06 F05.
Select year = 2006 funding only when calculating final 2007/08 factors with ILR 2006/07 F05.

Select if year = 2005.

Matches in fields from 0506 LAD to enable the identification of Key skills learning aims.

SORT CASES BY a09.

MATCH FILES

/FILE = *
/TABLE =\\LAD_0506.sav'
/BY a09.

create empty variables for each period and for the total year -.

Compute pn_fee_elem_KS = 0.
Compute pn_fee_elem_ORD = 0.
Compute pn_fee_elem_ENR = 0.
Compute fee_elem_KS = 0.
Compute fee_elem_ORD = 0.
Compute fee_elem_ENR = 0.

Compute pn_franchdi_KS = 0.
Compute pn_franchdi_ORD = 0.
Compute pn_franchdi_ENR = 0.
Compute franchdi_KS = 0.
Compute franchdi_ORD = 0.
Compute franchdi_ENR = 0.

Compute pn_corefund_KS = 0.
Compute pn_corefund_ORD = 0.
Compute pn_corefund_ENR = 0.
Compute corefund_KS = 0.
Compute corefund_ORD = 0.
Compute corefund_ENR = 0.

Compute pn_feerem_KS = 0.
Compute pn_feerem_ORD = 0.
Compute pn_feerem_ENR = 0.
Compute feerem_KS = 0.
Compute feerem_ORD = 0.
Compute feerem_ENR = 0.

Compute new_pn_fee_elem_KS = 0.
Compute new_fee_elem_KS = 0.
Compute new_pn_fee_elem_ORD = 0.
Compute new_fee_elem_ORD = 0.

Compute new_pn_franchdi_KS = 0.
Compute new_franchdi_KS = 0.
Compute new_pn_franchdi_ORD = 0.
Compute new_franchdi_ORD = 0.

***Flag Key skills ***.

Where Keyskills variable in learning aims database equals 1 or 2 compute the new variable Key_skill equals 1.

Compute key_skill = 0.
Do if any(a_kscode,"1","2").
Compute key_skill = 1.
end if.

value labels key_skill 0 'Not Key Skill learning aim'
1 'Key Skill learning aim'.

Flag Entitlement dummy aim.

When learning aim sequence equals 99 compute the new variable Enrichment = 1.

Compute Enrichment = 0.
Do if A05 = 99.
Compute Enrichment = 1.
End if.

value labels Enrichment 0 'Ordinary learning Aim'
1 'Enrichment Dummy Aim'.

Flag ordinary learning aims.

When learning aim sequence equals anything other than 99 compute the new variable ordinary_aim = 1.

Compute ordinary_aim = 0.
Do if A05 <> 99.
Compute ordinary_aim = 1.
End if.

value labels ordinary_aim 0 'Enrichment Dummy Aim'
1 'Ordinary learning Aim'.

*** For each period calculate fee element, Franchised discount, Programme funding and fee remission for the three following categories of records***.

*** Key skills learning aims only (KS), all genuine learning aims (ORD) and additional enrichment records (ENR)***.

Compute pn_fee_elem_KS = key_skill * fee_elem.
Compute pn_fee_elem_ORD = Ordinary_aim * fee_elem.
Compute pn_fee_elem_ENR = Enrichment * fee_elem.
Compute pn_franchdi_KS = key_skill * franchdi.
Compute pn_franchdi_ORD = Ordinary_aim * franchdi.
Compute pn_franchdi_ENR = Enrichment * franchdi.
Compute pn_corefund_KS = key_skill * corefund.
Compute pn_corefund_ORD = Ordinary_aim * corefund.
Compute pn_corefund_ENR = Enrichment * corefund.
Compute pn_feerem_KS = key_skill * feerem.
Compute pn_feerem_ORD = Ordinary_aim * feerem.
Compute pn_feerem_ENR = Enrichment * feerem.
end if.

*** For the whole year calculate fee element, Franchised discount, Programme funding and fee remission for the three following categories of records***.

*** Key skills learning aims only (KS), all genuine learning aims (ORD) and additional enrichment records (ENR)***.

Compute fee_elem_KS = key_skill * fee_elem.
Compute fee_elem_ORD = Ordinary_aim * fee_elem.
Compute ee_elem_ENR = Enrichment * fee_elem.
Compute franchdi_KS = key_skill * franchdi.
Compute franchdi_ORD = Ordinary_aim * franchdi.
Compute franchdi_ENR = Enrichment * franchdi.
Compute corefund_KS = key_skill * corefund.
Compute corefund_ORD = Ordinary_aim * corefund.
Compute corefund_ENR = Enrichment * corefund.
Compute feerem_KS = key_skill * feerem.
Compute feerem_ORD = Ordinary_aim * feerem.
Compute feerem_ENR = Enrichment * feerem.

Calculate revised fee element for Key skills only.

Compute new_pn_fee_elem_KS = pn_fee_elem_KS * (!new_perc_fee_elem / !current_perc_fee_elem).
Compute new_fee_elem_KS = fee_elem_KS * (!new_perc_fee_elem / !current_perc_fee_elem).

Calculate revised fee element for all genuine learning aims.

Compute new_pn_fee_elem_ORD = pn_fee_elem_ORD * (!new_perc_fee_elem / !current_perc_fee_elem).

Compute new_fee_elem_ORD = fee_elem_ORD * (!new_perc_fee_elem / !current_perc_fee_elem).

****Where applicable calculate new Franchised discount for Key skills only and for all genuine learning aims****.

Do if franchdi > 0.

Compute new_pn_franchdi_KS = pn_franchdi_KS - (pn_fee_elem_KS * !number3).

Compute new_franchdi_KS = franchdi_KS - (fee_elem_KS * !number3).

Compute new_pn_franchdi_ORD = pn_franchdi_ORD - (pn_fee_elem_ORD * !number3).

Compute new_franchdi_ORD = franchdi_ORD - (fee_elem_ORD * !number3).

end if.

***Calculate new programme funding for Key skills only and for all genuine learning aims ***.

Compute new_pn_corefund_KS = pn_corefund_KS + (pn_fee_elem_KS - new_pn_fee_elem_KS) + (pn_franchdi_KS - new_pn_franchdi_KS).

Compute new_corefund_KS = corefund_KS + (fee_elem_KS - new_fee_elem_KS) + (franchdi_KS - new_franchdi_KS).

Compute new_pn_corefund_ORD = pn_corefund_ORD + (pn_fee_elem_ORD - new_pn_fee_elem_ORD) + (pn_franchdi_ORD - new_pn_franchdi_ORD).

Compute new_corefund_ORD = corefund_ORD + (fee_elem_ORD - new_fee_elem_ORD) + (franchdi_ORD - new_franchdi_ORD).

***Calculate new fee remission funding for Key skills only and for all genuine learning aims ***.

Compute new_pn_feerem_KS = pn_feerem_KS * (!new_perc_fee_elem/current_perc_fee_elem).

Compute new_feerem_KS = feerem_KS * (!new_perc_fee_elem/current_perc_fee_elem).

Compute new_pn_feerem_ORD = pn_feerem_ORD * (!new_perc_fee_elem/current_perc_fee_elem).

Compute new_feerem_ORD = feerem_ORD * (!new_perc_fee_elem/current_perc_fee_elem).

Aggregate all variables to learner level.

Sort cases I01 I03.

AGGREGATE

/OUTFILE = *

/PRESORTED

/BREAK I01 I03

/pn_fee_elem_KS = sum(pn_fee_elem_KS)

/fee_elem_KS = sum(fee_elem_KS)

/pn_fee_elem_ORD = sum(pn_fee_elem_ORD)

/fee_elem_ORD = sum(fee_elem_ORD)

/pn_fee_elem_ENR = sum(pn_fee_elem_ENR)

/fee_elem_ENR = sum(fee_elem_ENR)

/pn_franchdi_KS = sum(pn_franchdi_KS)

/franchdi_KS = sum(franchdi_KS)

/pn_franchdi_ORD = sum(pn_franchdi_ORD)

/franchdi_ORD = sum(franchdi_ORD)

/pn_corefund_KS = sum(pn_corefund_KS)

/corefund_KS = sum(corefund_KS)

/pn_corefund_ORD = sum(pn_corefund_ORD)

/corefund_ORD = sum(corefund_ORD)

/pn_corefund_ENR = sum(pn_corefund_ENR)

/corefund_ENR = sum(corefund_ENR)

/pn_feerem_KS = sum(pn_feerem_KS)

/feerem_KS = sum(feerem_KS)

/pn_feerem_ORD = sum(pn_feerem_ORD)

/feerem_ORD = sum(feerem_ORD)

/pn_feerem_ENR = sum(pn_feerem_ENR)

/feerem_ENR = sum(feerem_ENR)

/fee_elem = sum(fee_elem)

/franchdi = sum(franchdi)

/achieve = sum(achieve)

/addsupp = sum(addsupp)

/corefund = sum(corefund)

/feerem = sum(feerem)

/new_pn_fee_elem_KS = sum(new_pn_fee_elem_KS)

/new_fee_elem_KS = sum(new_fee_elem_KS)

/new_pn_fee_elem_ORD = sum(new_pn_fee_elem_ORD)

/new_fee_elem_ORD = sum(new_fee_elem_ORD)

/new_pn_franchdi_KS = sum(new_pn_franchdi_KS)

/new_franchdi_KS = sum(new_franchdi_KS)

/new_pn_franchdi_ORD = sum(new_pn_franchdi_ORD)

/new_pn_corefund_KS = sum(new_pn_corefund_KS)

/new_corefund_KS = sum(new_corefund_KS)

/new_pn_corefund_ORD = sum(new_pn_corefund_ORD)

/new_corefund_ORD = sum(new_corefund_ORD)

/new_pn_feerem_KS = sum(new_pn_feerem_KS)

/new_feerem_KS = sum(new_feerem_KS)

/new_pn_feerem_ORD = sum(new_pn_feerem_ORD)

/new_feerem_ORD = sum(new_feerem_ORD)

/aims = n.

Calculate total of revised programme and fee remission for Key skills learning aims only.

Compute new_pn_corefund_feerem_KS = new_pn_corefund_KS + new_pn_feerem_KS.

Compute new_corefund_feerem_KS = new_corefund_KS + new_feerem_KS.

***Calculate combined totals of current Key skills and enrichment record programme funding ***.

Compute pn_corefund_KS_ENR = pn_corefund_KS + pn_corefund_ENR.

Compute corefund_KS_ENR = corefund_KS + corefund_ENR.

***Calculate combined totals of current Key skills and enrichment record fee remission funding ***.

Compute pn_feerem_KS_ENR = pn_feerem_KS + pn_feerem_ENR.

Compute feerem_KS_ENR = feerem_KS + feerem_ENR.

Calculate combined totals of current Key skills and enrichment record fee remission and programme funding.

Compute pn_corefund_feerem_KS_ENR = pn_Corefund_KS_ENR + pn_feerem_KS_ENR.

Compute corefund_feerem_KS_ENR = Corefund_KS_ENR + feerem_KS_ENR.

Calculate current programme and fee remission funding for enrichment records only.

Compute pn_corefund_feerem_ENR = pn_corefund_ENR + pn_feerem_ENR.

Calculate current programme and fee remission funding for Key Skills learning aims only.

Compute pn_corefund_feerem_KS = pn_corefund_KS + pn_feerem_KS.

Calculate the difference between the total of the current and the revised programme and fee remission funding for Key skills learning aims.

Compute pn_KS_difference = 0.

Do if pn_corefund_ENR > 0.

Compute pn_KS_difference = new_pn_corefund_feerem_KS - pn_corefund_feerem_KS.

End if.

Calculate the revised combined total of core and fee remission funding for Key skills and additional enrichment records.

Compute new_pn_corefund_feerem_ENR_KS = pn_corefund_feerem_ENR - pn_KS_difference + new_pn_corefund_feerem_KS.

Calculate maximum possible revised Fee remission total for key skill aims and dummy enrichment aim.

Compute feerem_total_ENR = !period_enrichment_amount * !new_perc_fee_elem.

Calculate revised per period maximum possible funding for dummy enrichment aim.

Compute pn_Max_new_total_ENR = 0.

Do if new_pn_corefund_feerem_ENR_KS > new_pn_corefund_feerem_KS.

Compute pn_Max_new_total_ENR = new_pn_corefund_feerem_ENR_KS - new_pn_corefund_feerem_KS.

End if.

Calculate revised annual maximum possible funding for dummy enrichment aim.

Compute Max_new_total_ENR = p1_Max_new_total_ENR + p2_Max_new_total_ENR + p3_Max_new_total_ENR.

Calculate revised annual franchised discount.

Compute new_franchdi_ORD = new_p1_franchdi_ORD + new_p2_franchdi_ORD + new_p3_franchdi_ORD.

Calculate possible revised fee remission funding for enrichment dummy aim.

Compute pn_Possible_Fee_rem = feerem_total_ENR - new_pn_feerem_KS.

Calculate revised per period fee remission funding for enrichment dummy aim.

Compute new_pn_feerem_ENR = 0.

Do if pn_Max_new_total_ENR > 0 and pn_Possible_Fee_rem > 0 and pn_corefund_ENR > 0.

Compute new_pn_feerem_ENR = min(pn_Max_new_total_ENR, pn_Possible_Fee_rem).

end if.

Calculate revised annual fee remission funding for enrichment dummy aim.

Compute new_feerem_ENR = new_p1_feerem_ENR + new_p2_feerem_ENR + new_p3_feerem_ENR.

Calculate revised per period core funding for enrichment dummy aim.

Compute new_pn_corefund_ENR = pn_Max_new_total_ENR - new_pn_feerem_ENR.

Calculate revised annual core funding for enrichment dummy aim.

Compute new_corefund_ENR = new_p1_corefund_ENR + new_p2_corefund_ENR + new_p3_corefund_ENR

Calculate total revised annual core funding.

Compute new_corefund = new_corefund_ORD + new_corefund_ENR.

Calculate total revised fee remission funding.

Compute new_feerem = new_feerem_ORD + new_feerem_ENR.

Aggregate to provider level.

AGGREGATE

/OUTFILE = *

/PRESORTED

/BREAK 101

/new_corefund = sum(new_corefund)

/achieve = sum(achieve).

Calculate provider level Achievement factors – _0607 for final and replace naming to _0708 for provisional.

Compute Achievement_factor_0607 = achieve / new_corefund.